University Institute of Intelligent Systems and
Numeric Applications in Engineering

Ph.D. Thesis

Accurate Estimation of Object Motion in Image Sequences

Nelson Monzón López

Las Palmas de Gran Canaria
April, 2017

UNIVERSIDAD DE LAS PALMAS DE GRAN CANARIA
Instituto Universitario de Sistemas Inteligentes
y Aplicaciones Numéricas en Ingeniería

EDUARDO M. RODRÍGUEZ BARRERA, CON DNI 437352427X, PROFESOR TITULAR DE UNIVERSIDAD Y SECRETARIO DEL INSTITUTO UNIVERSITARIO DE SISTEMAS INTELIGENTES Y APLICACIONES NUMÉRICAS EN INGENIERÍA (SIANI) DE LA UNIVERSIDAD DE LAS PALMAS DE GRAN CANARIA,

CERTIFICA

Que el Consejo de Doctores del Instituto Universitario de Sistemas Inteligentes y Aplicaciones Numéricas en Ingeniería (SIANI), en su sesión de fecha 27 de abril de 2017, tomó el acuerdo de dar el consentimiento para la tramitación de la Tesis Doctoral titulada *Accurate Estimation of Object Motion in Image Sequence*, presentada por el doctorando D. Nelson Monzón López, dirigida por el Dr. D. Javier Sánchez Pérez y el Dr. D. Agustín de la Nuez Salgado, propuesta para acceder a la mención de doctorado internacional, a la vista de la idoneidad y calidad de su contenido, interés y relevancia del tema a nivel internacional, y la realización de una estancia de más de un trimestre en un centro de investigación fuera de España.

Para que así conste, y a los efectos oportunos se expide el correspondiente certificado a 28 de abril de 2017.

UNIVERSIDAD DE LAS PALMAS DE GRAN CANARIA

Departamento: Instituto Universitario de Sistemas Inteligentes y Aplicaciones Numéricas en Ingeniería

Programa de Doctorado: Doctorado en Sistemas Inteligentes y Aplicaciones Numéricas en Ingeniería

Título de la Tesis

Accurate Estimation of Object Motion in Image Sequences

Tesis Doctoral presentada por D Nelson Monzón López

Dirigida por el Dr. Javier Sánchez Pérez

Codirigida por el Dr. Agustín Salgado de la Nuez

El/la Director/a, **El/la Codirector/a** **El/la Doctorando/a,**

Las Palmas de Gran Canaria, a 28 de Abril de 2017

A mi familia...

Agradecimientos

El desarrollo de una tesis doctoral supone un gran esfuerzo en el que han intervenido un gran número de personas. Aprovecho este momento para agradecer la ayuda y el enorme apoyo recibido durante este periodo. Sin ellos, no habría sido capaz de lograr todos los objetivos que me he fijado durante estos últimos años.

En primer lugar, agradecer a mis padres el enorme esfuerzo que han realizado para que sus tres hijos salgamos adelante y, en particular, financiar nuestros estudios universitarios. El dinero no lo es todo y no puedo más que destacar el enorme apoyo moral que nos han dado durante estos años. Este último año ha sido muy duro pero como familia siempre "tiramos hacia delante". También me gustaría agradecer a mi novia Ana que siempre me ha apoyado a conseguir mis metas.

Por otro lado, quiero expresar mi más sincero agradecimiento a los directores de la tesis, Javier Sánchez Pérez y Agustín Salgado de la Nuez, por su dedicación, orientación y continua supervisión de este trabajo de investigación. Sin la ayuda de ambos esta tesis no se habría conseguido. Han sido muchas horas de pizarra, revisiones e ideas para que todas nuestras publicaciones y este documento hayan podido salir adelante.

Durante todos estos años he podido disfrutar de un genial ambiente de

trabajo en el laboratorio de investigación CTIM en la Universidad de Las Palmas de Gran Canaria. Esto ha sido por la enorme ayuda y apoyo que me han brindado, tanto en cuestiones docentes como de investigación, varios miembros del centro: Agustín Trujillo, Miguel Alemán, Esther González, Julio Esclarín, Luis Gómez, Luis Mazorra y Luis Álvarez. También me gustaría agradecer a Karl Krissian que aunque ya no esté en el centro siempre será uno de sus miembros. Muchas gracias a todos.

Otra causa clara de este gran ambiente han sido los distintos compañeros que he tenido en el laboratorio, que hicieron que fuera algo más que una sala con ordenadores. Quisiera recordarlos: Dani, Pedro, Kilian y Airam. Se echa de menos la hora de la fruta... Dani, tal y como escribiste en la dedicatoria de tu tesis... ¡lo logramos!

Quisiera agradecer también a Jean-Michel Morel por acogerme en el seno de su grupo de investigación, CMLA en la ENS CACHAN, durante mi estancia predoctoral. Su inestimable ayuda y "pasión" por la investigación reproducible han sido una guía para los objetivos de esta tesis. Quisiera agradecer a los miembros del equipo CMLA por su apoyo y los ratos compartidos: Miguel Colom, Tristan Dagobert, Pablo Arias, Gabriele Facciolo y Enric Meinhardt. También me gustaría agradecer a Luis Álvarez y Luis Mazorra su enorme ayuda para poder financiar la estancia.

Por último, quisiera agradecer a las instituciones que han financiado parte de los trabajos realizados en el contexto de esta tesis.

Contents

Abstract

The aim of this thesis is to contribute in the calculation of variational optical flow methods. This is a basic topic in the field of computer vision that pursues the accurate estimation of the displacement experienced by the objects present in a video scene. In particular, this dissertation is focused on two main themes: (i) we study the influence of temporal information compared to traditional spatial variational methods; (ii) we analyze several strategies for the preservation of flow discontinuities and propose alternatives to overcome this problem. Nowadays, these two issues remain unsolved and we consider them important for finding better optical flow fields. According to the enormous increment of the automation in the industry, the use of artificial intelligence and computer vision techniques in particular becomes more important. In this context, it is relevant to find automatic and well founded numerical methods to interpret moving scenes from image sequences.

The document is divided in five chapters. In the first chapter we introduce the problem and give a guideline of this document. In the second, we study the most relevant works from the state-of-the-art that fits with the problems that we are dealing. Besides, we present several issues closely related with the context of this thesis, like standard datasets for optical flow studies or reproducible research.

In the third chapter, we propose a spatio-temporal variational method for the consistent estimation of large motion fields. Our focus is on the development of realistic temporal coherence models that are suitable with current spatial models. The aim of this work is to explore ways of temporal coherence that takes into account the non-continuity of large motion fields. In this sense, we propose three main contributions: (i) a nonlinear flow constancy assumption, similar to the nonlinear brightness constancy assumption, (ii) a nonlinear temporal regularization approach; (iii) an anisotropic diffusion operator based

on the Nagel-Enkelmann operator.

The chapter four presents an implementation of the spatial and temporal approaches of the Brox *et al.* method and compare their main features. We also study various solutions using grayscale and RGB images from recent evaluation datasets to verify the color benefits in motion estimation.

Finally, we analyze several strategies for the discontinuity-preserving problem in variational methods. Our analysis includes the use of tensors based on decreasing functions, which has shown to provide good results. We observe that this strategy is normally unstable if the function is not well controlled introducing instabilities in the computed motion field. Our conclusions lead us to propose two alternatives to overcome these drawbacks: (i) a simple approach that combines the decreasing function with a minimum isotropic smoothing; (ii) a fully automatic strategy that adapts the diffusion depending on the image features. It looks for the best parameter configuration that preserves the important motion contours and avoid instabilities.

Our contributions have been tested on standards benchmark databases that are in common use in optical flow.

Resumen

El objetivo de este tesis es contribuir en el cálculo de flujo óptico. Este es un tema importante dentro del campo de la visión por computador que persigue estimar de manera precisa el desplazamiento que experimentan los objetos presentes en una escena de vídeo. En la figura 1 se ofrece un ejemplo de lo que significa la estimación del flujo óptico. Utilizamos para ello la secuencia de *Alley 1* que pertenece a la base de datos de *Sintel*[1] [Butler12].

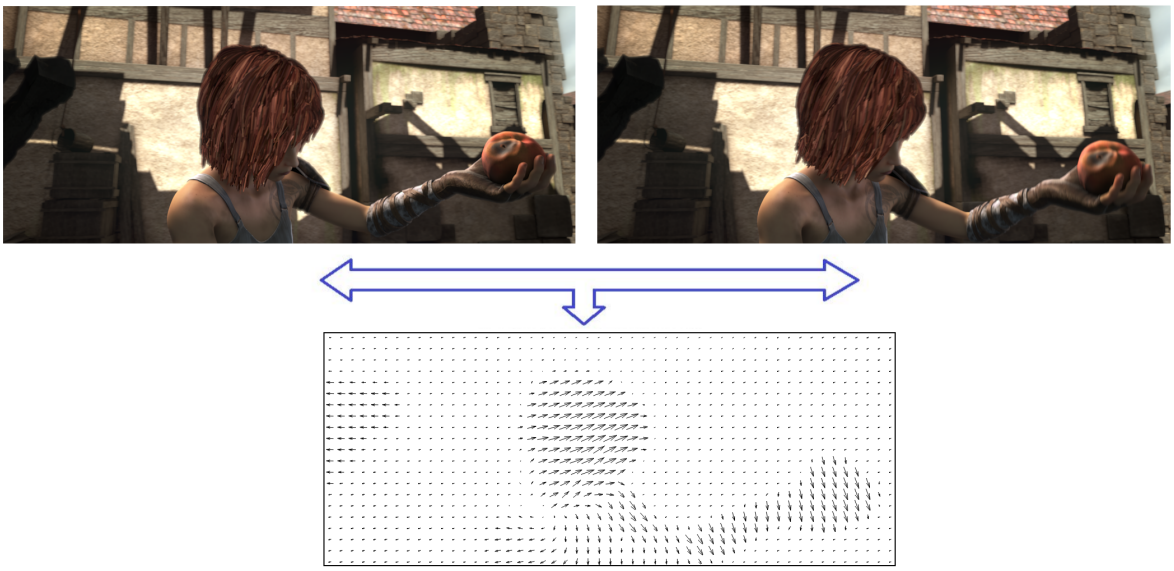


Figura 1: Ejemplo de flujo óptico. En la parte superior vemos dos *frames* consecutivos de las secuencia *Alley 1* que pertenece a la base de datos de *Sintel*. En la imagen inferior, vemos un campo vectorial que representa el movimiento presente en esta secuencia.

[1] http://sintel.is.tue.mpg.de/

La importancia del cálculo del flujo óptico es muy elevada ya que sirve como base para un amplio número de aplicaciones dentro del campo de la visión artificial, así como para resolver muchos tipos de problemas, como pueden ser: la reconstrucción 3D, el guiado de robots, la compresión de vídeo, el registrado de imágenes médicas, el análisis de imágenes meteorológicas y muchas otras. Toda esta variedad de aplicaciones nos da una idea de la importancia que tiene el flujo óptico dentro de esta área de investigación. A medida que aumenta la automatización en la industria, se hace más importante la utilización de técnicas de inteligencia artificial y de visión por ordenador en particular.

Este trabajo de tesis tiene **dos objetivos principales**: (i) Estudiar la influencia que tiene la información temporal comparada con los métodos variacionales de tipo espacial; (ii) Analizar varias estrategias para la preservación de discontinuidades en el flujo y proponer alternativas para solventar este problema.

A pesar de más de treinta años de estudios de flujo óptico, estos temas siguen sin estar del todo resueltos y, además, en el mundo actual es cada vez más relevante encontrar métodos numéricos que puedan emular de manera automática la habilidad humana de interpretar escenas en movimiento a partir de secuencias de imágenes. Por este motivo, consideramos relevantes las propuestas expuestas en este trabajo.

El documento se divide en cinco partes: En la primera parte introducimos el problema y comentamos brevemente la guía de este documento. En el segundo capítulo estudiamos los trabajos más relevantes del estado del arte que mejor encajan con los problemas que tratamos. Además, presentamos varias cuestiones muy relacionadas con el contexto de esta tesis, tales como bases de datos estándar en estudios de flujo óptico o investigación reproducible. En el tercer capítulo proponemos un método variacional de tipo espacio-temporal. Nuestra intención es contribuir en modelos de coherencia temporal que sean aplicables en modelos espaciales. El objetivo de este trabajo es explorar modelos temporales que tengan en cuenta la no continuidad en desplazamientos largos. En este sentido, proponemos tres principales contribuciones: (i) un término que ofrece una asunción no lineal de constancia en el flujo, (ii) un nuevo esquema de regularización que trata con flujos ópticos no continuos, (iii) Un esquema anisotrópico de difusión basado en el operador de Nagel-Enkelmann.

En cuarto lugar, presentamos una implementación propia de las aproximaciones espacial y temporal del método de Brox *et al.* y comparamos sus principales

características. Además estudiamos varias soluciones usando imágenes en escala de grises y RGB que pertenecen a bases de datos recientes para verificar si la información de color beneficia a la estimación de movimiento.

Finalmente, analizamos varias estrategias orientadas al problema preservación de discontinuidades en métodos variacionales de flujo óptico. Nuestro análisis incluye el uso de tensores de difusión basados en funciones decrecientes que han demostrado buenos resultados. El uso de funciones decrecientes suelen introducir inestabilidades en el campo de flujo computado. Nuestras conclusiones nos han llevado a proponer dos alternativas que permiten solventar esta desventaja: (i) Una aproximación que combina una función decreciente pero acompañada de un valor constante, con el objetivo de asegurar que se produzca siempre un mínimo de difusión isotrópica, (ii) un método que busca la mejor configuración de parámetros que preserve los contornos del flujo al mismo tiempo que evite las inestabilidades. Este último esquema ofrece un sistema automático que regula su influencia a partir de las caracterısticas de la imagen.

Todas las contribuciones expuestas en esta tesis han sido testeadas en bases de datos estándar ampliamente utilizadas en estudios de flujo óptico.

Conclusiones

A continuación, detallamos el trabajo realizado y resumimos las conclusiones obtenidas durante los distintos capítulos. Hemos contribuido en el estudio del flujo óptico de dos maneras:

- Tratamiento de la información temporal en métodos de flujo óptico.

  En el capítulo 3 propusimos un método variacional que hace uso de coherencia temporal para encontrar campos de flujo más continuos. Este método combina una asunción no lineal de constancia en el flujo en el modelo de energía además de un esquema no lineal de regularización del flujo. La primera relaciona convenientemente los campos de flujo en diferentes instantes de tiempo mientras que la segunda produce flujos continuos en el tiempo. Este también incluye un operador robusto de difusión anisotrópica basado en la técnica de Nagel-Enkelmann y una técnica de

flujo inverso que permite encontrar correspondencias en el tiempo. Denominamos a este método como TCOF.

Hemos realizado pruebas en secuencias sintéticas y reales para poder comparar los resultados con su contrapartida temporal. Observamos que, cuando usamos secuencias reales, TCOF ofrece campos de flujo continuos en el tiempo y que el ruido se reduce respecto a la solución espacial. Esto es un comportamiento interesante y refleja que las propuestas de este capítulo encajan correctamente con los términos no lineales estándar de constancia del brillo y del gradiente. TCOF ofrece importantes mejoras, especialmente cuando el método se enfrenta a desplazamientos largos.

- Estrategias eficientes para tratar con el problema de preservación de discontinuidades.

  En el capítulo 4, realizamos un análisis exhaustivo del método de Brox *et al.* Observamos buenos resultados ofreciendo flujos continuos y robustos frente a *outliers*. Sin embargo, también crea formas redondeadas en los límites del flujo y normalmente estos no coinciden con los contornos de los objetos de la imagen. Posteriormente, introdujimos el esquema espacial dentro de un esquema multicanal para evaluar los posibles beneficios. Concluimos que la mejora es evidente y que el coste computacional es razonable y está justificado.

  La información de color mejora la preservación de discontinuidades pero aún produce formas redondeadas. Esto ocurre porque la regularización depende del flujo y no se utiliza información de la propia imagen. Introducir una función decreciente en el término de suavizado puede ser una posible solución a este problema. Por esta razón, estudiamos varias técnicas orientadas a la preservación de discontinuidades en el capítulo 5. En particular, nos centramos en el uso de funciones decrecientes y tensores de difusión anisotrópicos.

  Observamos que esta estrategia ofrece precisión en los contornos del flujo pero, sino escogemos cuidadosamente los parámetros, aumentan considerablemente los errores en el resultado. Esto es porque un parámetro equivocado introduce inestabilidades en el flujo óptico. Además, detectamos que este problema ocurre en la gran mayoría de las secuencias y no es anecdótico.

  En este sentido, propusimos dos estrategias para solventar este problema. La

primera consiste en combinar la función decreciente con una pequeña constante que asegure siempre un mínimo de suavizado impidiendo que se cancele completamente el término de regularización. La segunda es una aproximación automática que adapta la difusión dependiendo del histograma de gradientes de la imagen. Comparando su evolución del error con respecto al parámetro de discontinuidad vemos que reducen sensiblemente el problema de las inestabilidades y que ambas propuestas presentan un buen comportamiento en las discontinuidades del flujo.

La primera propuesta es muy estable y ofrece soluciones muy competitivas. Sin embargo, creemos que la aproximación automática es mejor ya que normalmente consigue los mejores resultados sin intervención externa. En nuestra opinión, esta característica hace que la segunda propuesta pueda ser interesante para aplicaciones reales.

Chapter 1

Introduction

Computer vision encompasses a wide research field that pursues the comprehension of the data captured in digital images or videos. Related to this, a broad variety of high-level applications within the field of artificial vision, such as robotic guidance, objects tracking, 3D scene reconstruction, augmented reality, image medical registration, video compression and many others, require a low-level process that allows a precise motion recovery from their surrounding environment.

In this context, optical flow estimation is very useful because it provides consistent information of the apparent displacement of the pixels in a video sequence. Once we have calculated our solution, we can determine the movement of the objects through the sequence. Owing to its importance and multiple uses, the optical flow problem has become a major theme in computer vision and it is the objective of this dissertation.

In figure 1.1, we observe an example of the meaning of optical flow estimation by using the sequence of *Alley 1*, which belongs to the *Sintel* Dataset[1] [Butler12]. The idea is that, if we add into the first image the movement described by the flow field for each pixel, we should obtain the following image.

Variational optical flow methods are among the most widely used techniques in the literature if accuracy is the main objective. Typically, these approximations obtain their solutions as a minimization of an energy functional that allows obtaining dense flow fields, which means that the displacement values are provided for the whole domain. Another interesting property of the variational methods is that they are transparent, in the sense that all the assumptions on the image data and the solution are explicitly formulated in the underlying energy functional. There are no intermediate or post processing steps that question the consistency of the whole approach. Moreover, the use of a joint minimization framework allows a solid mathematical integration of all desired assumptions.

[1] http://sintel.is.tue.mpg.de/

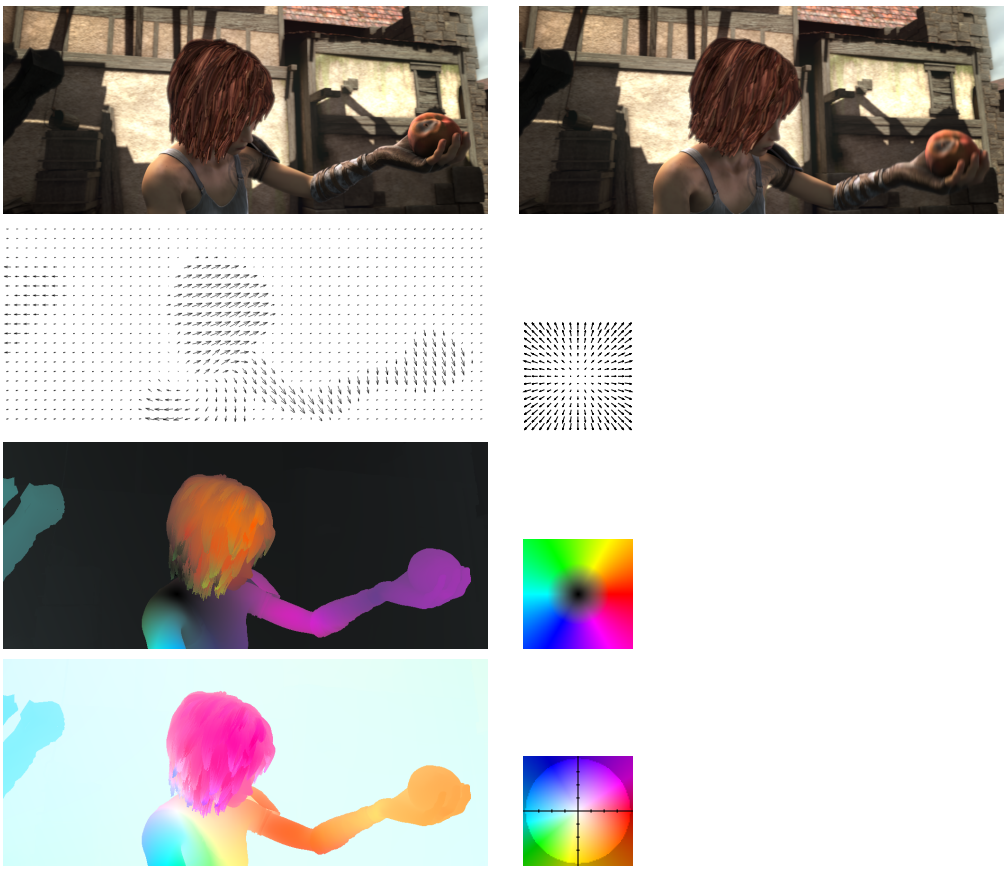


Figure 1.1: Representation of the optical flow. At the top, it is shown a pair of images of the *Alley 1* sequence from the *Sintel* Dataset. In the pictures below, there are depicted flow fields representing the motion present in this sequence by using three different strategies. The scheme for the representation is at the right of each solution.

This energy is expressed as a weighted sum of a data and a smoothness terms. The first puts in correspondence pixels from consecutive images under a premise, commonly known as *brightness constancy assumption*, that the intensity of the objects remains constant through the sequence. The second imposes a constraint on the continuity of the flow that ensures that our solution is unique assuming that neighboring pixels present a similar motion. Otherwise, the optical flow arise to an *ill-posed* problem in the sense that there may exist an infinity of solutions for the matching of two images.

The minimization of this energy functional typically generates a system of partial differential equations (PDE) whose solution must fulfill several restrictions imposed on the model. This system results into a scheme that is solved using techniques of numerical analysis and whose solution can be implemented in a standard programming language. Although there exist other possibilities like multi grid schemes, in our case, we solve our

system of equations by using the SOR (*Successive Over-Relaxation*) method. This is an iterative procedure that solves linear system of equations, resulting in faster convergence.

In order to estimate large displacements and avoid irrelevant local minimum values, we embed the optical flow method in a multi-scale strategy as in previous works like [Anandan89, Battiti91, Luettgen94, Bornemann96, Enkelmann88, Mémin02]. This allows us to create a coarse-to-fine structure for solving the system of equations in each scale, starting at the coarsest scale, to get successive approximations of the optical flow for refining the solution at finer scales.

We represent the flow fields by using three different strategies. The first one consists in using directly a vectorial representation to describe the relative motion while the other two strategies use color schemes. Each color denotes a different direction and their intensities the magnitude of the moving objects (see figure 1.2). In the document, we call them as IPOL and *Middlebury* color schemes, respectively. The chromaticity of the *Middlebury* color scheme is clearer than the IPOL representation, which is interesting for saving costs when printing. However, the IPOL color scheme is more intuitive, since the black color represents no motion.

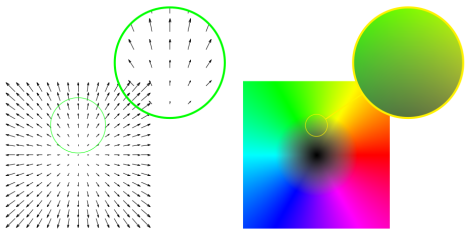


Figure 1.2: Relation between the color scheme and the vectorial field for the optical flow representation.

Despite of more than thirty years of optical flow studies, there are still some limitations in current variational methods like, for instance, occlusions handling, that arise when a portion of the image is visible at one frame but not in its successive, the estimation of large displacements or the preservation of discontinuities in the displacement field. From these issues, the intention of this manuscript is to contribute in two themes: (i) the influence of temporal information compared to traditional spatial variational methods; (ii) Efficient strategies for improving the preservation of flow discontinuities.

1.1 Main contributions

This section describes the main contributions that have been published in the context of this thesis:

- **Nonlinear Temporal Coherence Models for Optical Flow Estimation [Sánchez13f, Sánchez13e, Sánchez13d, Sánchez13c, Sánchez15]**

  We present nonlinear variational models for the coherent estimation of motion fields. The aim of this work is to explore new ways of temporal coherence that takes into account the non-continuity of large motion fields. In this sense, we propose two main contributions: a nonlinear flow constancy assumption, similar in spirit to the nonlinear brightness constancy assumption, which conveniently relates flow fields at different time instants; and a nonlinear temporal regularization approach, that fosters smooth flow fields in time. These can be easily combined with traditional spatial models, consistently linking the flow information in the spatial and temporal dimensions.

- **Robust Optical Flow Estimation [Sánchez13b, Monzón14b, Sánchez12]**

  We describe an implementation of the variational method proposed by Brox *et al.*, in 2004, which yields accurate optical flows with low running times. It has several benefits with respect to the method of Horn and Schunck: it is more robust to the presence of outliers, produces piecewise-smooth flow fields and can cope with constant brightness changes. It also generalizes the use of continuous L^1 functionals, which help mitigate the effect of outliers and create a Total Variation (TV) regularization. Additionally, it introduces a simple temporal regularization scheme that enforces a continuous temporal coherence of the flow fields.

- **Regularization Strategies for Discontinuity-Preserving Optical Flow Methods [Monzón16a, Monzón16b, Monzón13, Monzón14a, Sánchez14]**

  Our objective is to study several strategies for the preservation of flow discontinuities in variational optical flow methods. We analyze the combination of robust error functions and diffusion tensors in the smoothness assumption. Our study includes the use of tensors based on decreasing functions, which has shown to provide good results. However, it presents several limitations and usually does not perform better than other basic approaches. It typically introduces instabilities in the computed motion fields in the form of independent *blobs* of vectors with large magnitude. We propose two alternatives to overcome these drawbacks: first, a simple approach that combines the decreasing function with a minimum isotropic smoothing; second, a method that looks for the best parameter configuration that preserves the important motion contours and avoid instabilities.

1.2 Acknowledgements

The work developed in this dissertation is one of the main research lines of the group 'Análisis de la Imagen e Ingeniería del Software' (AIIS) led by Javier Sánchez Pérez,

that belongs to the research center: Centro de Tecnologías de la Imagen (CTIM) at the University of Las Palmas de Gran Canaria (ULPGC). Special thanks to Prof. Jean-Michel Morel for hosting me during a research internship at the group 'Centre de Mathématiques et de Leurs Applications' (CMLA) in the École Normale Supérieure de Cachan (Paris).

The work has been partially supported by the Spanish Ministry of Science and Innovation through the research project TIN2011-25488, the University of Las Palmas de Gran Canaria grant ULPGC011-006, the BPIFrance and Région Ile de France, in the framework of the FUI 18 Plein Phare project, the European Research Council (advanced grant Twelve Labours) and the Office of Naval research (ONR grant N00014-14-1-0023).

1.3 Outline

- Chapter 2, *The Optical Flow Problem*:

  This chapter describes the state of the art concerning the problem of optical flow estimation. First, we summarize the main issues dealing with optical flow methods. Second, we mention the most relevant methods of the literature, describing some of the most innovative techniques that have emerged in recent years that have served as reference to other methods developed later. This is followed by related works regarding to the main challenges that we address in this dissertation: Temporal coherence and motion contours preservation.

  We explain the main characteristics of the most widely used datasets in the optical flow literature. We describe the sequences used in the experiments of this thesis, as well as the metrics used for obtaining quantitative results.

- Chapter 3, *Nonlinear Temporal Coherence in Optical Flow Methods*:

  Applying temporal coherence in variational methods favors finding more continuous flow fields in time. Besides, the occlusion handling could be ameliorated by using information from successive frames. In this sense, in chapter 3, we propose several contributions: (i) nonlinear flow constancy assumption; (ii) a novel nonlinear flow regularization scheme that can deal with non-continuous optical flows; (iii) an anisotropic diffusion operator based on the Nagel-Enkelmann operator. The results show that a nonlinear temporal formulation of the flow field provided accurate motion fields.

- Chapter 4, *Robust Optical Flow Estimation*:

  We study spatial and temporal approaches of the Brox *et al.* proposal [Brox04]. Then, we improve the results of this method by including color information in its energy functional. We observe that the flow discontinuities are typically degraded by the continuous temporal regularizer; therefore, the spatial approach is still a good

option due to its more simplicity and faster solutions respect to the discontinuity preserving problem in optical flow methods.

- Chapter 5, *Robust Discontinuity-Preserving Optical Flow Methods*:

  The preservation of the motion boundaries is one of the main issues that we analyze in this dissertation. Once we have developed and deeply studied the behavior of the classical TV-L^1 approach of the Brox method, we observed that this type of methods typically creates rounded effects at flow boundaries, which usually do not coincide with object contours. A simple strategy to overcome this problem consists in inhibiting the diffusion at high image gradients. Here, we first introduce a general framework for TV regularizers in optical flow and relate it with some standard approaches. Then, we propose efficient strategies to improve the discontinuity-preserving problem.

- Chapter 6, *Conclusions and Future Work*:

  In this chapter, we present the final conclusions of the thesis, discussing the problems encountered during its development and highlighting the contributions made to the literature in all the works collected in this document. Then, we enumerate some tasks that due to their complexity and lack of time (and budget) have been left for the future.

- Appendices:

  - Appendix I: *Estimation of the backward flow.*

    The temporal method exposed in chapter 3 requires backward flow for its correct performance. In this appendix, we present algorithms for computing the inverse optical flow in image sequences.

  - Appendix II: *Details of implementation of the ROF method.*

    Here, we present the pseudo-codes and a brief explanation of the parameters of the implementation presented in chapter 4.

  - Appendix III: *Details of implementation of the RDPOF method.*

    In this appendix, we present several details and the parameters of the discontinuity preserving strategies proposed in chapter 5.

  - Appendix IV: *Work flow for the RDPOF demo.*

    In this section, we explain the work flow and details of the demo associated with the RDPOF demo. This is a demo that belongs to the IPOL journal.

Chapter 2

The Optical Flow Problem

This chapter describes the state-of-the-art concerning to the optical flow problem. We review some of the most important methods in the literature and discuss successful techniques used in recent years and which has been a source of inspiration for this thesis. We also present the sequences and the datasets used in the different experiments, the reproducibility research that has lead our work and the error metrics on which we rely to quantitatively evaluate our solutions.

The optical flow is a vector field that puts in correspondence the pixels between two images. Given a continuous image sequence, we represent the images as an application $I : (x, y, t) \rightarrow I(x, y, t)$ where (x, y) is the spatial coordinate of the image and t is the time. The motion vector is defined as $\mathbf{w}(x, y, t) = (u(x, y, t), v(x, y, t), 1)^\top$ and represents the horizontal and vertical displacements, respectively.

If we want to find correspondences between the pixels, we need to assume that some property of the image is invariant through the time. In this sense, the intensity of the pixels is a value that indicates the amount of luminance reflected by the surface of the objects. There are many models to represent different types of surfaces. One of the simplest is the *Lambertian* model. On a Lambertian surface the apparent brightness is the same for all view directions. A pixel belonging to this type of surface will maintain the same intensity value in the complete sequence. This is known as *brightness constancy assumption*. For this reason, if we assume that the surfaces of the objects are Lambertian, we can represent this invariance as

$$I(x + u, y + v, t + 1) - I(x, y, t) = 0, \tag{2.1}$$

where t and $t + 1$ are the present instant and its successive in the image sequence. This is not always fulfilled in the real scenes where the luminance conditions can easily change due to many circumstances.

The latter expression (2.1) is non-lineal but, if we assume that the image sequence

varies smoothly in space and time, we can avoid this non-linearity by performing a first-order Taylor expansion. Then, we obtain the well-known *optic flow constraint* (OFC)

$$I_x u + I_y v + I_t = 0, \tag{2.2}$$

where I_x, I_y are the first order derivatives in x and y. Another property that we will use in this thesis is the assumption that the color does not vary in time.

The OFC is not enough to estimate uniquely the optical flow because is one equation with two unknowns. This ambiguity is known as the *aperture problem*. In figure 2.1 is shown, from left to right, a line at time instants t and $t+1$, respectively. We can only know the displacement of the normal component in the direction of the gradient of the pixels of the line (red arrow). We can not be sure if that line has also shifted in the direction of the component orthogonal to the direction of the gradient.

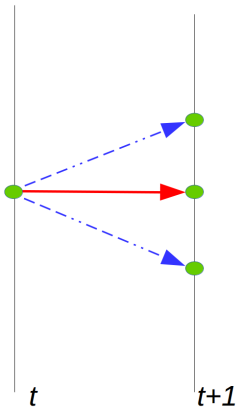


Figure 2.1: Example of the *aperture problem*.

The work of Horn and Schunck [Horn81] solves this issue by imposing an additional constraint assuming smoothness in the flow over the whole image. This restriction is known as smoothness (or regularization) term. The combination of both constancy and smoothness assumptions into a single formulation gives us the energy functional

$$E_{HS}(\mathbf{w}) = \int_\Omega \underbrace{(I_x u + I_y v + I_t)^2}_{\text{Constancy}} dx + \alpha \int_\Omega \underbrace{\left(|\nabla u|^2 + |\nabla v|^2\right)}_{\text{Smoothness}} dx. \tag{2.3}$$

The solution is obtained directly by minimizing this energy. Based on the existing mathematical theory if this energy meets a number of conditions the minimization process will allow to reach the global minimum and ensure the uniqueness of the solution. The parameter α determines the influence of the regularization in the functional. This was one of the first variational methods proposed and it is one of the most cited works in the optical flow literature.

2.1 Challenges in Optical Flow Methods

Despite of more than thirty years of variational optical flow studies, there still exists important questions and issues to solve. Some of these are the following:

- **Occlusions**: The occlusions problem arises when a portion of the image is visible at one frame but not in its successive. The movement itself, either of the camera or any of the objects, typically produces that part of the scene is hidden from the view of the camera. The occlusions create discontinuities in the optical flow which generates the violation of some of the restrictions imposed. The detection of occlusions improves the estimation of optical flow at the edges of objects. We see an example in figure 2.2.

Figure 2.2: Example of occlusion in an image. We observe many occlusions that hide the cars of the scene due to the different lampposts, traffic lights and poles in the *Rheinhafen* sequence.

- **Preservation of motion contours**: Flow boundaries are normally associated with the contours of the objects in the scene. However, the converse is not true, since adjacent objects, moving in the same direction, may belong to the same optical flow region, with no motion discontinuities between them. The problem becomes more challenging since it is difficult to differentiate between object contours and textures. A simple mechanism for inducing the boundaries of the flow field from the objects is not easy.

- **Large motions**: The presence of large displacements makes more difficult the correlation between a pair of successive images and creates non-continuous motions

Figure 2.3: Example of motion contours. We see in the image at the right a representation of the flow edges of a moving car by using the IPOL color scheme.

in time. This affects how to model regularity conditions in order to create smooth flow fields.

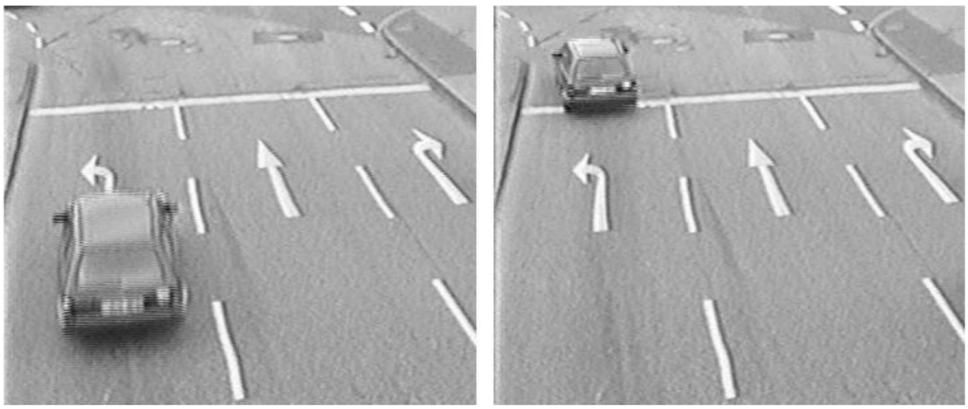


Figure 2.4: Example of a large displacement. The position of the car between the image on the left and its consecutive on the right represents a large displacement.

- **Velocity model**: Natural objects follow physical motion models in realistic scenes. Assumptions of inertial displacements, where velocity and acceleration determine the trajectory of objects, should be considered to provide smooth coherence flow fields in time.

- **Temporal continuation**: The optical flow can be incrementally computed from previous estimates. It is not reasonable to start from scratch with every new frame. This makes us think that the first optical flow should be computed in a different

way than the rest of flows. This pose a problem of *initialization* and *temporal continuation*.

- **Redundancy**: Normally, most of the information in the frames does not vary too much in short period of times. Perhaps we should take advantage of the intrinsic redundancy for creating better matching criteria.

  By using the temporal dimension, we come across problems that are different from the spatial dimension.

- **Frame rate**: The temporal resolution is given by the frame rate of the camera. This is completely different to the spatial resolution, which is given by the size of the images. The motion continuity depends on the speed of the objects. We use figure 2.5[1] as example of the frame rate.

Figure 2.5: Frame rate example

We also find two more problems with the frame rate: On the one hand, frames can be lost due to a poor camera performance; on the other hand, it is not assured that frames are captured with a constant frame rate.

- **Change of context**: In large sequences, the optical flow method have to cope with situations when the video changes completely from scene or a rapid rotation of the camera.

[1] http://www.mediacollege.com/video/frame-rate

- **Matching criteria**: The brightness constancy assumption is easily violated in real sequences. In fact, it is quite difficult that the intensities remain constant through a real sequences due to defects of the camera, imperfections of the lens, dust in the air or other factors. Even a cloud in the sky can vary the luminance in the scene. The optical flow methods usually incorporate techniques that correct or attenuate this effects such as, robust penalization or pre-processing algorithms before the calculation for reducing these type of problems. On the other hand, using multi-channel schemes can associate the pixel information with various brightness intensities. This helps when dealing with variations of the luminance.

2.2 Optical Flow Methods

In this section, we present some of the most important approaches in the optical flow literature. Some of them have been a source of inspiration for the present thesis. We comment their main contributions. For simplicity, we use the same notation being I_1 and I_2 the input images defined in a continuous domain $\Omega : (x, y) \rightarrow R$, where $x = (x, y)$ are the coordinates of the pixels.

Horn-Schunck [Horn81]

The Horn and Schunck method uses the OFC equation (2.2) and a regularization term that solves the *aperture problem*. Equation (2.3) shows the Horn and Schunck functional. This energy model uses quadratic functionals in both terms. This assumes that the image noise and the flow derivatives are expected to follow a Gaussian distribution. A direct consequence of this kind of functionals is that the method is very sensitive to the presence of noise and the computed flow fields are very smooth.

Since this proposal, many approaches have arisen inspired by their ideas in the form of new regularization terms like Nagel-Enkelmann [Nagel86], Alvarez *et al.* [Alvarez99], Uras *et al.* [Uras88], Schnörr [Schnörr94], Weickert [Weickert97, Weickert98] or the inclusion of more invariants in the data term like in Brox *et al.* [Brox04].

Lucas-Kanade [Lucas81]

Lucas-Kanade is one of the most representatives approaches that belongs to the *local* methods. This type of optical flow methods use the information obtained from the neighborhood around a pixel to estimate its motion. Lucas-Kanade calculates the displacement by minimizing the optical flow equation around a window centered on a pixel and assuming that the motion is constant in this neighborhood.

$$E(u,v) = \sum_{(x,y)\in N} W^2(x,y)\left(I_x u + I_y v + I_t\right)^2, \tag{2.4}$$

where $W(x,y)$ is the windows centered on the pixel (x,y) and N is the neighborhood.

One of the biggest drawback of this type of methods is that it is only possible to detect the movement in those areas where there are variations in the image. In homogeneous zones where there may be movement this one is not detectable. Therefore, the fields of displacement are not dense. Some authors have used more functions than pixel intensities to prevent the problem from being poorly conditioned, such as Mitiche et al. [Mitiche87], Wohn et al. [Wohn83]. Others have used differential operators such as Tretiak-Pastor [Tretiak84] or Campani-Verri [Campani90].

Nagel-Enkelmann [Nagel86]

The energy model proposed by Nagel-Enkelmann was a great contribution in the field of optical flow. Their model is similar to Horn-Schunck but it includes an anisotropic smoothing term. This term varies the direction of diffusion according to the gradient of the image. In regions where the gradient is small it acts isotropically and in those where the gradient is high the diffusion is anisotropic along the edges.

In [Nagel86], the method uses only two frames but, some years later, it was created a new spatio-temporal approach of this model in [Nagel90a]. The following functional describes the spatial version of this method.

$$E(\mathbf{w}) = \int_\Omega \left(I_1(\mathbf{x}) - I_2(\mathbf{x}+\mathbf{w})\right)^2 d\mathbf{x} + \alpha \int_\Omega tr(\nabla \mathbf{w}^\top D \nabla \mathbf{w}) d\mathbf{x}, \tag{2.5}$$

The diffusion operator (D) proposed by Nagel and Enkelmann [Nagel86] preserves discontinuities of the images in the flow field. This is a continuous diffusion operator that allows for anisotropic diffusion along the edges and isotropic smoothing in homogeneous regions. This operator is based on a projection matrix in the orthogonal direction to the gradient of the image,

$$D(\nabla I) = \frac{\nabla I^\perp \nabla I^{\perp T} + \lambda^2 \mathbf{Id}}{\|\nabla I\|^2 + 2\lambda^2},$$

with $\mathbf{Id}$ the identity matrix. λ determines the value of the gradient from which the anisotropy is activated.

Since this operator depends on the gradient of the image, the main drawback is that the motion discontinuities tend to proliferate, especially in textured regions. A similar proposal was presented in Alvarez *et al.* [Alvarez00] using an scale-space strategy to obtain a coarse-to-fine scheme that benefits the optical flow dealing with large displacements.

Black-Anandan [Black96]

Robust functions provide methods that reduce the influence of failures in the input data. In optical flow, many strategies use robust functions to remove outliers. In 1996, Black and Anandan [Black96] proposed a method that robustified the OFC and the regularization. Their idea was to reduce the problems due to violations in the brightness and spatial smoothness assumptions caused by multiple motions.

$$E(u,v) = \sum_{m \in M} \left[\lambda_D \Psi_D \left(I_x u + I_y v + I_t, \eta \right) + \lambda_S \left[\sum_{z \in Z} \Psi_R \left(u_z - u_n \right) + \sum_{z \in Z} \Psi_R \left(v_z - v_n, \eta \right) \right] \right] \tag{2.6}$$

with $\Psi(s) = \log(1 + \frac{1}{2}\left(\frac{s}{\eta}\right)^2)$ the Lorentzian norm. They assumed that $\Psi_D = \Psi_R$ in their experiments. η is a control parameter for the robust functions. $M = \{m_1, m_2, ..., m_{n^2} | \forall w, 0 \leq i(s_w), j(s_w) \leq n-1\}$ defines a site of the image with size $n \times n$ pixels. Z represents the north, south, east and west neighbors of M.

Brox *et al.* [Brox04]

The method of Brox *et al.* [Brox04] uses a data term composed of two invariants: constant gradient and the brightness constancy assumption. Their smoothing was a robust version of the one proposed by Horn-Schunck. This is based on differentiable L^1 functionals that creates piecewise-smooth flow fields. This keeps a better spatial coherence between both unknowns of the optical flow, at the expense of creating rounded effects at flow discontinuities. The functional reads as

$$\begin{aligned} E(\mathbf{w}) = & \int_\Omega \Psi \left(\left(I_1(\mathbf{x}) - I_2(\mathbf{x}+\mathbf{w}) \right)^2 + \gamma \left(\nabla I_1(\mathbf{x}) - \nabla I_2(\mathbf{x}+\mathbf{w}) \right)^2 \right) d\mathbf{x} \\ & + \alpha \int_\Omega \Psi \left(|\nabla u|^2 + |\nabla v|^2 \right) \mathbf{dx}, \end{aligned} \tag{2.7}$$

with $\Psi(s^2) = \sqrt{s^2 + \epsilon^2}$ and $\epsilon := 0.001$ a small constant. This method (both spatial and spatio-temporal versions) is deeply studied in chapter 4.

Zach *et al.* [Zach07]

The method of Zach *et al.* [Zach07] minimizes a data term using the L^1 norm and a regularization term that uses the total variation of the flow. This proposal can be considered a robust version of the classical Horn and Schunck method [Horn81] that

allows discontinuities in the flow field and is more robust against noise by changing the quadratic factors. They linearize the data term as

$$\rho(\mathbf{u}) = \nabla I_2(\mathbf{x} + \mathbf{u}^0) \cdot (\mathbf{u} - \mathbf{u}^0) + I_2(\mathbf{x} + \mathbf{u}^0) - I_1(\mathbf{x}) = 0, \tag{2.8}$$

with $\mathbf{u}^0$ a close approximation to $\mathbf{u}$. Then, they minimize the functional by introducing the following convex relaxation

$$E_\theta(\mathbf{u}, \mathbf{v}) = \int_\Omega |\nabla u_1| + |\nabla u_2| + \frac{1}{2\theta} |\mathbf{u} - \mathbf{v}|^2 + \lambda |\rho(\mathbf{v})| . \tag{2.9}$$

Setting θ to a very small value forces the minimum of E_θ to occur when $\mathbf{u}$ and $\mathbf{v}$ are nearly equal, reducing to the original energy E, defined in equation (3.1). The interest of this relaxation is that E_θ can be minimized by alternatively fixing one of $\mathbf{u}$ or $\mathbf{v}$, and solving for the other variable.

1. Fixed $\mathbf{v}$, solve
$$\min_{\mathbf{u}} \int_\Omega |\nabla u_1| + |\nabla u_2| + \frac{1}{2\theta} |\mathbf{u} - \mathbf{v}|^2 . \tag{2.10}$$
2. Fixed $\mathbf{u}$, solve
$$\min_{\mathbf{v}} \int_\Omega \frac{1}{2\theta} |\mathbf{u} - \mathbf{v}|^2 + \lambda |\rho(\mathbf{v})| . \tag{2.11}$$

The first sub-problem fits the total variation denouncing model of Rudie-Sher-Fate mi [Rudin92], which can be solved by Chamomile's duality-based algorithm [Chambolle04]. The second sub-problem does not depend on spatial derivatives of $\mathbf{v}$, so it can be solved point-wise by thresholding. In 2012, Ballester *et al.* [Ballester12] proposed a variational method that joints the previous proposal with an occlusion detector based on the divergence of the flow.

Optical Flow in Harmony [Zimmer11]

The work of Zimmer *et al.* [Zimmer11] presented the spatial and spatio-temporal approaches of a method that combines diffusion tensors with robust functions in order to avoid over segmentation in the flow field. A similar diffusion scheme is used for studying the regularization at motion contours in chapter 5. This approach includes a data and a regularization term, as follows:

$$E(\mathbf{w}) = \int_\Omega \left(\mathcal{D}(\mathbf{w}) + \alpha \cdot \mathcal{R}(\nabla \mathbf{w}) \right) \mathbf{dx}. \tag{2.12}$$

Their data term uses constraint normalization, both brightness and gradient constancy assumptions and HSV color representation. They also use separate robust penalization for each color channels like in [Bruhn05a]. This allows robustness under outliers and varying illumination conditions. The attachment term reads as

$$\mathcal{D}(\mathbf{w}) := \sum_{c=1}^{3} \Psi_{\mathcal{D}} \left(\mathbf{w}^T \tilde{\mathbf{J}}_1^c \mathbf{w} \right) + \gamma \left(\sum_{c=1}^{3} \Psi_D \left(\mathbf{w}^T \tilde{\mathbf{J}}_2^c \mathbf{w} \right) \right), \tag{2.13}$$

where $\tilde{J}$ is the motion tensor that mixes both brightness and constancy assumptions like in Brox *et al.* [Brox04] and the summatory is due to the use of the three HSV channels.

The smoothness term uses information from the motion tensor in the data term. Their diffusion tensor resembles a robust variant of the Nagel-Enkelmann operator, with decreasing functions to mitigate the diffusion across the boundaries of the objects. The diffusion is performed unconditionally along the isocontours while in the gradient direction it is modulated by a robust function.

$$\mathcal{R}(\nabla \mathbf{w}) := \Psi_{\mathcal{R}} \left(u_{r_1}^2 + v_{r_1}^2 \right) + (u_{r_2}^2 + v_{r_2}^2), \tag{2.14}$$

with $\Psi(s^2) = \frac{1}{\lambda^2} log\left(1 + \lambda^2 s^2\right)$ [Perona90] and (r_1, r_2) are eigenvalues of a structure tensor.

Sun-Roth-Black Weighted Median Filter [Sun10a]

This work formalizes median filtering as non-local term that integrates information over large spatial neighborhood. They included this term, that combines median filtering and L^1 minimization [Li09], into a variational method that uses the Charbonnier [Charbonnier97] function to robustified their functional. This strategy adaptively preserves motion details.

Other Methods

There exist several strategies for dealing with large displacements, such as the use of coarse-to-fine schemes [Bergen92, Alvarez00], descriptor matching, nearest neighbors (NN) or approximate nearest neighbors (AAN), that are aimed at solving this problem. Other approaches, such as graph-cuts or belief-propagation, rely on discrete optimization to find global solutions.

Brox and Malik [Brox11a] proposed a descriptor matching into a variational method to estimate large motions of small structures. This is a technique for estimating arbitrarily large displacement. Their method do not improve the accuracy but, they support the coarse-to-fine warping strategy in avoiding local minima. They split the non-convex

optimization problem into a series of tractable subproblems. The method in [Lei09] proposes a similar solution based on region matching and discrete optimization. The source image is represented as a tree of regions. Coarse-to-fine scheme based on region trees. The segmentation is carried out using mean-shift filtering. However, it searches in a neighborhood around the current estimation and it may lose small structures motions.

Other strategy that has been recently used in optical flow is the PatchMatch filter [Lu13]. This is an algorithm that quickly finds correspondences between small patches in an image [Barnes09, Barnes10]. Recently, the authors of [Hu16] propose to combine a coarse-to-fine scheme with this PatchMatch, in order to detect large displacements at the same time that it enables small structures to be detected at the finest scales through random search. This is used as sparse matching in [Revaud15]. The procedure consists in three steps: an initial sparse matches (Deep matching [Weinzaepfel13]); an interpolation algorithm and a variational method for the optical flow. Their experiments showed that Deep Matching provides better results compared to AAN in optical flow. The work presented in [Bailer15] uses a correspondence field approach (Flow Fields) and an outliers filtering with the method in [Revaud15].

Belief propagation has a long tradition in stereoscopic vision [Sun03] and can be also used in optical flow [Yedidia03]. It resembles the regularization process: Every region receives the candidate flows of its neighbors, with their corresponding probabilities. This information allows to update the probability of the candidates in each region.

2.3 Temporal Coherence in Optical Flow Methods

The seminal work of Horn and Schunck [Horn81] proposed a basic idea about temporal coherence. They proposed to compute the flow field between the first two images and then use this estimate as initial guess for the following frame. This is a simple predicting technique, that introduces one of the desirable ideas for temporal coherence methods: the *temporal continuity* [Black94] of the calculations that should result in less computations. This idea is exploited later in some other works, where the flow is predicted and updated according to the observed data.

This is the strategy followed with Kalman filtering approaches like in [Singh91]. In this work the filter was used to incrementally compute the optical flow. The process is divided in prediction and update phases: during the prediction phase, the previous flow is extrapolated to an intermediate state vector and, in the update phase, the measurements are mixed with the predicted state to create a final state. The method is based on a correlation approach, so the solutions are expected to be sparse. The Kalman filter allows to reduce the uncertainty and improve the flow field in subsequent predictions.

The most widely used strategy is a very simple formulation based on the regularization of the temporal derivative. This has shown to be effective when the motion field is

continuous, as has been originally demonstrated in [Weickert01]. Although this is a simple idea, it is not realistic for actual optical flow applications.

The main handicap of using the temporal derivative is the implicit assumption that the motion field must be continuous. This poses a sort of incongruence with respect to the nonlinear attachment of the data, which is specially designed for large displacements. The use of robust functions may mitigate the effect of temporal smoothing, since it avoids taking into account the outliers from the continuous model. When the temporal derivative is coupled with the spatial derivatives, it may also mitigate the effect of the spatial regularization. Isolating the temporal term helps solving this problem, although it is only effective in the presence of small motions.

Another method to propose a time formulation were Black and Anandan [Black91]. They proposed a temporal coherence constraint based on the assumption of constant accelerations. The flow field was predicted from the averaged acceleration and the previous estimation of the optical flow. In the energy model they introduced a new temporal term based on the attachment of the flow to the estimated flow field that incorporates the information of acceleration. Nagel [Nagel90b] also proposed a temporal method based on an extension of his spatial diffusion operator as a 3D diffusion matrix.

More recently, some authors have generalized the use of regularization schemes based on the flow temporal derivative. This was analyzed in [Weickert01] and later used in other related approaches, e.g. [Brox04], [Bruhn05b] or [Papenberg06]. In these cases, the temporal information is coupled with the spatial gradient in the form of a non-quadratic 3D smoothing operator. Empirical results have demonstrated that the temporal information should have a lower penalty weight than the spatial counterpart. This is probably due, in part, to the fact that, while the spatial information is continuous and the gradient can be numerically approximated with enough precision, the temporal information is not in general continuous, specially in the presence of large displacements. The use of temporal information has demonstrated to provide important improvements with respect to the spatial formulation. As stated in [Weickert01] it is also more robust against noise and produces smoother flow fields. However, it treats the temporal information in the same way as the spatial one.

One of the firsts to propose a nonlinear temporal coherent model were Salgado and Sánchez [Salgado06]. In that case, they proposed to separate the temporal constraint from the spatial regularization and not to use the temporal derivative. They showed that it has a stabilizing effect on the estimated errors through the sequence. They devised the need to introduce the backward flow to improve the result in the whole sequence. Here we show that it is not necessary to explicitly introduce the backward flow in the energy model since it naturally appears during the minimization process. Additionally, we propose a nonlinear temporal regularization scheme.

In general, the taxonomy of temporal optical flow methods can be divided in three different groups: *causal models*, that compute the optical flow in an incremental way,

using the information of past and present images, e.g. [Black91] or [Chin94]; *anti-causal models*, that only use information from future and present frames; and *non-causal models* (or *acausal*), that use past and future information to compute the optical flow at the current frame, e.g. [Weickert01], [Brox04]. This taxonomy is the traditional classification used in digital signal processing for the design of filters. The former is suitable for real-time applications, like robot guidance, and the latter is useful for off-line processes, like video compression. Apparently, anti-causal systems does not seem to be applicable for the estimation of optical flow. It is reasonable that the non-causal models should provide higher accuracies than the causal models, since more information is available. In this sense, the new model that we propose in the present work is classified as a non-causal model. In general, we may regard the non-causal systems as a generalization of temporal coherence models.

Energy-based methods are among the most accurate methods in the literature. The *Middlebury* database [Baker07b] holds a ranking on outstanding optical flow methods.

In Xu *et al.* [Xu10] was presented a balance function between the brightness and gradient constancy terms, a mechanism to select flow candidates from multiple displacements using the SIFT features detector in each scale, and a simple occlusion handling technique. The use of SIFT detectors has also been proposed in other works, e.g. [Brox10], for estimating the motion of small scale structure.

In the work by Sun *et al.* [Sun10a], they propose to use a modified version of the median filtering during the optimization process. The median filtering is adapted to the image discontinuities. This allows also to respect the motion of small scale structures at the same time that the motion better fits the shape of the figures. A different approach is the other work by Sun et al [Sun10b] . They propose a probabilistic model in where the motion is separated in layers. These layers are ordered in depth and the occlusions are determined reasoning on this ordering.

2.4 Multichannel Optical Flow Methods

Optical flow methods are usually based on grayscale images. In fact, some methods convert the original color image into grayscale in order to obtain faster execution times and to decrease the complexity of the algorithm. However, the grayscale values are susceptible to slight changes in brightness, which often appear in real scenes.

This situation could be prevented by using color images due to the fact that these ones associate the information of their pixels with various brightness intensities. Even, the color information may avoid the use of additional constraints in the flow calculation [Golland97, Ohta89] and contains more photometric information that can be useful against shadows, shading and image specularities [Barron02, vdW04, Mileva07].

Color images build the scenes as the matching of different features stored by multiple channels of information. There exist several models depending of their characteristics and number of channels. Probably, the most commonly known is the RGB, that uses three channels for combining red, green and blue light components to reproduce a broad array of colors. The grayscale images are very similar but they only use a single channel of information for the luminance.

The YUV and HSV models store the image information in three channels as cylindrical-coordinate representations. Both schemes decompose the information in two channels for the chrominance and the brightness. Their difference is related to the color representation: while the YUV system uses an orthogonal three dimensional space of the color plane, the HSV model describes it as a vector in polar form. The three channel UCS color model gives an uniform chromaticity space based on measurements of human color perception. This measure is denoted by the Euclidean distance that corresponds linearly to the color perception or changes in the intensity.

We can also find other alternatives like, for instance, the CMYK model that composes the scene by mixing four channels of information (cyan, magenta, yellow and black) or satellite images. This last model can be defined as a visual representation captured by multiple electromagnetic spectrum channels of a sensor mounted on an artificial satellite.

During the years, various optical flow works have proposed the use of color information in their models. In this sense, Ohta [Ohta89] presented a method that calculates its solutions directly from an image pixel without imposing any other assumption [Fennema79, Horn81]. Therefore, the model derived multiple conditions from the multi-channel information of a single image point. Later, Markandey and Flinchbaugh [Markandey90] proposed a numerical scheme that uses the visible and the infra-red spectrum from a multi-spectral image.

Seven years later, Golland and Bruckstein [Golland97] presented two methods that increase the robustness of their energy models based on the color information. The first one assumes brightness conservation under motion considering a multi-channel image as a set of three different grayscale images. The second introduced the idea of *color conservation* under the premise that its components (not only the luminance) are conserved. Their experiments confirmed that, in regions with strong gradient, good solutions can be obtained, whereas in regions of uniform chrominance these methods failed. This work used RGB, normalized RGB and HSV images to evaluate their proposals.

In 2001, Barron and Klette [Barron01] reviewed the previous proposals [Ohta89, Golland97] and developed multi-channel extensions of the models of Horn and Schunck [Horn81] and Lucas and Kanade [Lucas81] methods. Their experiments concluded that the color is clearly beneficial in the computation of optical flow. Besides, they observed that a multi-channel scheme does not increase excessively the running times. In [Barron02], the authors reasserted their conclusions but determining, additionally, that the saturation channel worsen the motion fields.

Andrews and Lowell [Andrews03] compared the previous results of Horn and Schunck [Horn81] and Lukas and Kanade [Lucas81] with YUV and UCS color spaces. They also developed new implementations of [Golland97, Barron02] for achieving lower computational costs.

Weijer and Grevers [vdW04] based their proposal on the dichromatic reflection model of Schafer [Shafer85] for deriving photometric invariances against shadows, shading and specularities. Furthermore, their model incorporated a reliability measure for dealing with the instabilities that appear on the flow because of the photometric invariants. Their idea was to achieve robust motion fields, even under severe luminance conditions and noisy images, from the photometric information that the color provides.

In 2007, Mileva *et al.* [Mileva07] showed that a variational approaches improve their solutions against realistic luminance conditions by replacing the classical brightness assumption with photometric invariants. A few years later, Zimmer *et al.* [Zimmer11] proposed a robust data term against outliers and variable luminance. Among other features, it uses the HSV color space with separate robustification for each channel. This idea is interesting because the HSV representation offers distinct levels of photometric invariance, so we can use the most confident channel in our estimation. In 2010, Bin *et al* [Bin10] presented an approach that computes their solutions by extending the typical optical flow constraints. Their intention was to deal with movements like rotations, deformation objects motion and light variations.

Álvarez *et al.* [Álvarez08] contributed in the use of multi-channel images in optical flow. This work studied how to combine different channels of satellite images like, infra-red or intensity, to follow the evolution of the clouds.

A comparison of motion fields obtained with grayscale and color optical flow models is given in chapter 4. This leads us to extend our contributions into multi-channel schemes to improve our results.

2.5 Determining Motion Discontinuities

One of the main problems in variational optical flow methods is the preservation of motion boundaries that are normally associated with the contours of the objects in the scene. However, when several overlapping objects move in the same direction, with the same magnitude, they belong to the same optical flow region, with no motion discontinuities between them. This is an important challenge because a poor discontinuities detection means that our optical flow does not fit correctly with the reality of the scene. This is quite problematic if our optical flow will be used in applications that require a good definition of the scene like, for instance, 3D-reconstruction.

Since the seminal work of Horn and Schunck [Horn81], many optical flow approaches

have proposed different strategies for dealing with the problem of discontinuities. In 1986, Nagel and Enkelmann [Nagel86] proposed that the regularization process should be steered by a diffusion tensor that depends on the image. Proesmans *et al.* [Proesmans94] introduced the anisotropic scheme of Perona and Malik [Perona90], based on decreasing functions that inhibit the diffusion at high image gradients, in the estimation of optical flow. They also introduced a symmetric coherence model that helps to detect the discontinuities of the flow.

On the other hand, Rudin *et al.* [Rudin92] proposed to minimize the Total Variation (TV) of an image with an attachment to the original image. This leads to a diffusion equation that reduces the image noise, yielding sharp edges. This TV scheme was introduced in optical flow by Cohen [Cohen93]. Black and Anandan [Black93, Black96] mixed the previous ideas by establishing the relation between robust statistics and anisotropic diffusion. They also extended the use of robust functions to the whole energy terms, turning their method more robust against outliers. They showed that this strategy deals with image noise at the same time that it preserves flow edges.

Some improvements on the Perona and Malik model are given in Black *et al.* [Black98]. This establishes the relation between this kind of anisotropic diffusion processes, robust statistics and the minimization of energy functionals. It generalizes the use of robust functions in order to deal with outliers. A review on different strategies for diffusion filtering in image regularization and restoration can be seen in [Weickert97, Weickert98]. The author introduces the theory underlying the use of diffusion tensors in image filtering, e.g., the structure tensor [Bigun87]. Another source of inspiration for discontinuity preserving in optical flow is related with the bilateral filtering, introduced by Tomasi and Manduchi [Tomasi98]. In this case, the idea is to regularize an image using the information of the pixels that are near the actual position and have similar intensities or colors.

The method by Álvarez *et al.* [Alvarez99] introduces a decreasing function to inhibit the smoothing at image contours. Nevertheless, they did not use any robust function in the data term, so it is more sensitive to image noise. Aubert *et al.* [Aubert99] explicitly propose to use an L^1 functional in the data term and any justification function for smoothing. Alvarez *et al.* [Alvarez00] uses the Nagel-Enkelmann diffusion tensor, together with a nonlinear brightness formulation and a linear scale-space for the estimation of large displacements.

The generalization in the use of differentiable L^1 functions was proposed in [Brox04, Bruhn05b] and subsequent works. In fact, this has already been proposed before, e.g., in [Aubert99], but the former introduced a term based on the attachment of the image gradients, which is invariant to constant brightness changes. On the other hand, non-differentiable L^1 functions have also been used in Zach *et al.* [Zach07] that relies on a dual formulation, which yields a very efficient numerical scheme. The work in [Wedel09b] increases the robustness to illumination changes using the textural part of the images,

somehow similar to the gradient term of Brox *et al.* [Brox04]. Although the former two approaches are similar, they provide different results, as can be seen in the on-line works [Sánchez13b] and [Sánchez13a], respectively.

Some examples in the use of diffusion tensors with robust functions are given in [Zimmer09, Zimmer11]. In this case, the authors introduce a motion tensor in the data term and a regularization tensor in the smoothness term, which are designed in a similar way, taking into account not only the variation of image intensities but also the variation of the image gradients. The latter tensor uses a quadratic penalization for the diffusion along the contours, while a Perona-Malik diffusivity is used for mitigating the diffusion across flow edges.

TV-L^1 methods have several drawbacks: (i) they create rounded shapes near the borders or corners of the objects; (ii) typically, the edges are dislocated and usually do not coincide with the image contours; (iii) they produce staircase effects, yielding piecewise but planar motion regions. The first two inconveniences are due to the fact that the regularization process does not depend on the image information but on the flow field. In order to avoid these, some methods have introduced decreasing functions in order to stop the diffusion at image boundaries. This idea originally comes from [Alvarez99] and has recently been used in several methods, such as in [Wedel09a, Xu10]. It has also been used in Werlberger *et al.* [Werlberger09], where a diffusion tensor steers the regularization in the direction of the image gradient and its orthogonal direction. It incorporates a decreasing function in the direction of the gradient, so it inhibits the smoothing across edges.

The most important problem of these inhomogeneous diffusion schemes is that they easily produce instabilities in the computed flow fields. Depending on the value of the parameters, the method may become ill-posed if the smoothing term is canceled. Most of the aforementioned strategies assign an empirical value to these parameters. Unfortunately, the parameters that better preserve discontinuities are those that risk to produce instabilities.

One way to avoid the ill-posed problem is to introduce a small constant for ensuring a minimum isotropic behavior, like in Monzón *et al.* [Monzón14a] or Ayvaci *et al.* [Ayvaci12]. However, the value of this constant depends on the regularization factor, so that if this is large, then discontinuities will not be respected. In chapter 5, we show that this strategy outperforms the basic approach.

The idea of bilateral filtering has been introduced in optical flow by and Yoon and Kweon [Yoon06] and Xiao *et al.* [Xiao06]. In this case, the authors propose to regularize the flow field depending on the proximity and similarity of the intensities and flow values. In fact, this has to be seen as an extended trilinear filtering. They also used the information of occlusions to manipulate the range of the filters. Bilateral filtering has been used more recently, in combination with a TV-L^1 approach, in [Werlberger10].

There exist other strategies for improving the definition of discontinuities. For instance, some authors propose to compute the optical flow at the same time that the objects in the scene are segmented. In this way, the segmentation provides more information about the edges. Some examples in this line are [Mémin98, Sun10b, Sun12, Unger12]. In [Sun10a], the authors combine median filtering and bilateral filtering, in a post-processing step, to improve the flow field at edges. Other approaches mix TV-L^1 strategies with descriptor matching, such as [Brox11b, Xu10, Palomares16], which estimate the motion of some sparse features that are later introduced in the optimization process. This allows to deal with small moving objects. Moreover, since some features are typically associated with edges, this may help to define discontinuities. The methods based on segmentation, post-filtering or feature based matching, are difficult to analyze, especially concerning the problem of discontinuity preserving. These usually mix different aspects that affect the definition of flow boundaries, from standard robust regularization approaches to other *ad-hoc* processes.

The problem of discontinuities has not been solved yet. Introducing a simple mechanism for inducing the boundaries of the flow field from the objects is not easy. Moreover, the problem becomes more challenging since it is difficult to differentiate between object contours and textures, if we rely on the information of the image gradients. In this regard, this is one of the main objectives that we want to tackle in this dissertation.

2.6 Datasets

Typically, a research process comprises several states like: the rise of an idea (or hypotheses), a review of the most competitive methods in the literature that fits with your research interest, developing your hypotheses and, once your method is stable, carry out a deeply evaluation of the advantages and disadvantages of your results for obtaining conclusions.

In the field of optical flow, this last step implied collecting images that contain the challenges that your method tries to solve. In this sense, we need a complete dataset to fulfill these challenges. These ones usually include a ground truth or true flow (both names are correct) that represents exactly the motion present between a pair of consecutive images. This true flow helps to measure how close is your solution respect to the best possible flow field using different error measures such as the *Average Angular Error* or the *Average End-point Error* (see section 2.8).

Yosemite with Clouds

The original *Yosemite*[2] sequence was developed by Lynn Quam. It contains fifteen

[2]ftp://ftp.csd.uwo.ca/pub/vision

pixels with a size of 316×252, that represents a synthetic scene of the national park of Yosemite (Sierra Nevada, California). The camera simulates an aerial navigation that enters into the park. There exist two versions of this sequence: *Yosemite with Clouds* and *Yosemite*. The difference is that, in the second, the clouds have been artificially removed.

In some of our experiments, we use the complete version, where the clouds motion presents an horizontal displacement of two pixels per frame and some lighting variations appear in these clouds. The sequence mixes translational and divergent motions. The first and last columns of figure 2.6 depict the sixth and seventh frames of both versions of this sequence, respectively. In the center, we show the corresponding true flows that represents the displacement using the IPOL color scheme.

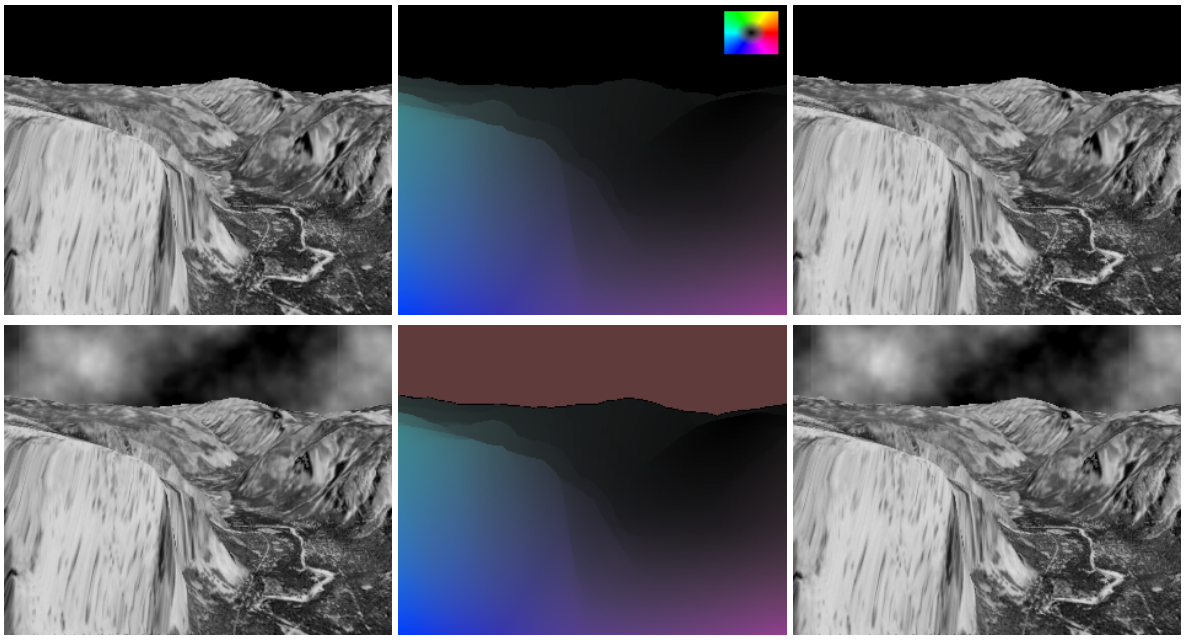


Figure 2.6: Frames 6 and 7 of *Yosemite* and *Yosemite with Clouds* sequences at the first and second rows, respectively. The center column shows their corresponding true flows. The motion is represented with the color scheme showed in the first row.

The Middlebury benchmark database

Probably, this is the most widely used dataset in optical flow studies during the last years. The *Middlebury* benchmark database[3] [Baker07b] offers several synthetic and realistic sequences for testing the methods.

Its synthetic scenes have been generated using computer graphics that creates several challenges like, fine textures, occlusions or large displacement. Each sequence provides a

[3]http://vision.middlebury.edu/flow/

true flow for the frame 10 to 11. Nowadays, this dataset is still one of the main challenges that an optical flow method must overcome to achieve the interest of the scientific community. Figure 2.7 shows some typical examples of this dataset. In particular, from top to bottom, we see the sequences of *RubberWhale*, *Grove2*, *Urban2* and *Hydrangea*. The center row shows the ground of each sequence.

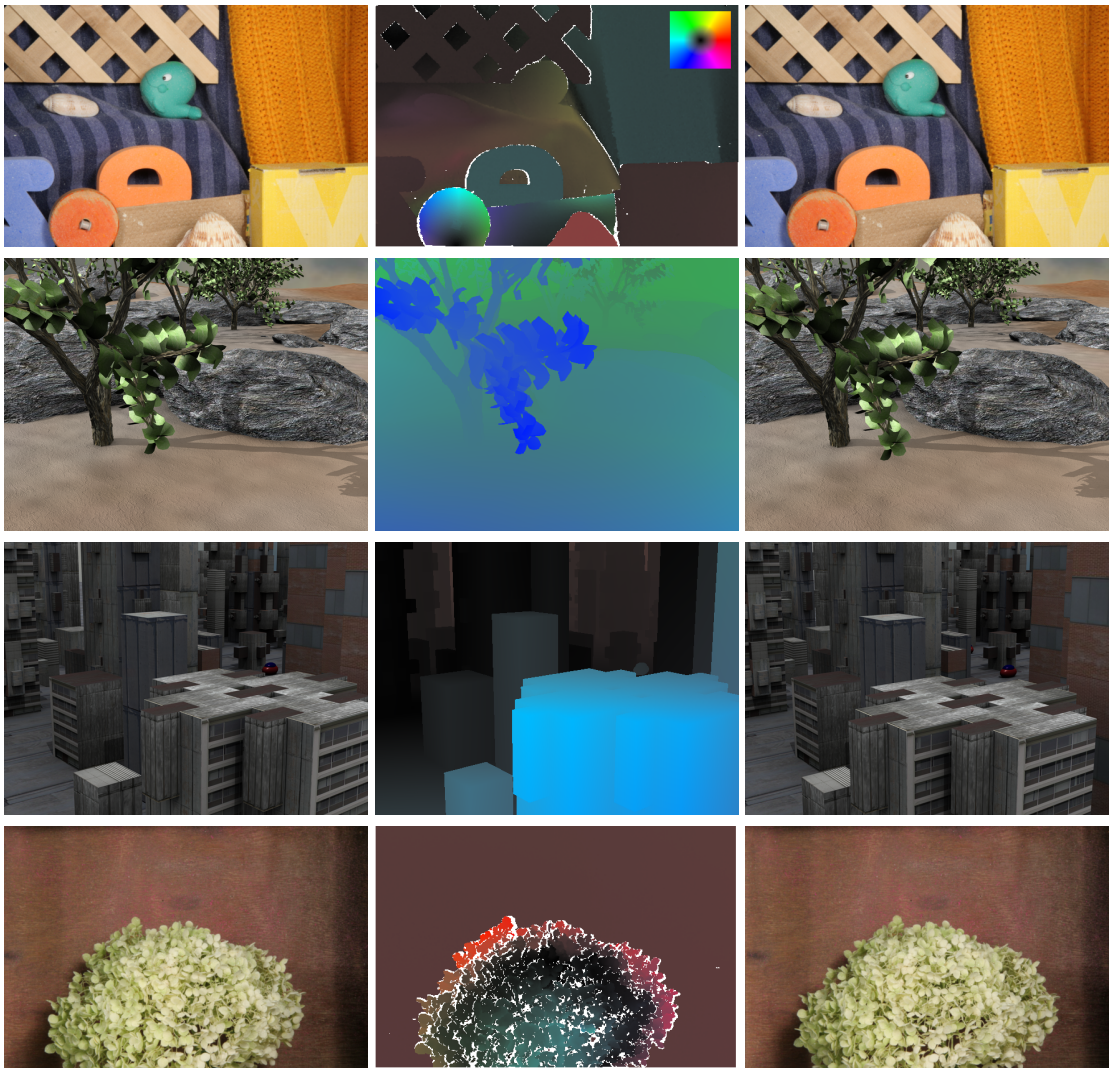


Figure 2.7: *Middlebury* benchmark dataset. From top to bottom, the first and third columns show consecutive images from the sequences of *RubberWhale*, *Grove2*, *Urban2* and *Hydrangea*, respectively. The central column depicts the corresponding true flows for every pair of images. The motion is represented with the color scheme showed in the first row.

MPI-Sintel dataset

Sintel, is a Dutch short-film premiered on September 27, 2010 at the Netherlands Film Festival that belongs to the adventure and fantasy genre. It was directed by Colin Levy and is the third short film produced by the Blender Foundation.

In 2012, some sequences of this film were used to create the *MPI-Sintel* Flow dataset[4] [Butler12]. It is a complex benchmark database to train and evaluate optical flow methods. As in the *Middlebury* dataset, *Sintel* provides true flows for every pair of images of each sequence to construct a scientific dataset. It allows that our algorithms face challenges like long-range motion, motion blur, defocus blur, atmospheric effects, multi-frame analysis and non-rigid motion. Generally, each sequence has approximately 20 frames. Some of them can be sen in figure 2.8. In chapters 4 and 5, we use these sequences for testing the methods with more complex scenes. The proposals presented in chapter 5 has been evaluated in the official web of *Sintel*. Figure 2.8 depicts some frames (and the corresponding ground truth) for the test sequences of *Alley 1*, *Ambush 4*, *Bamboo 2*, *Bandage 1* and *Shaman 2*.

Geometric sequences

Next, figure 2.9 shows several frames from a dataset developed by the AIRS group. The sequences are composed with geometric figures, such as rectangles, squares or stars, translating over a textured background. The motion of the figures is uniform with a displacement of 15 pixels per frame, while the background remains stationary except in the small sequence depicted at the first row. In this case, the background moves 3 pixels in the same direction. The most important features are its translational motion and a large displacement. These are simple sequences with a strong gradient variations at the objects boundaries.

Real-World sequences

In order to test several of our proposals with real-world data, we use the sequences of *Karl-Wilhelm-strasse*, *Rheinhafen*, *Ettlinger-Tor*, *MovingArm*, *ULPGC-Car* and *Taxi*. We show several frames of each sequence in figure 2.10.

The scene of *Karl-Wilhelm-strapless* belongs to a dataset with several traffic scenes created by Nagel[5] that consists in 1500 gray value frames with a size of 768×576. In this sequence, some vehicles are moving in different directions and speed, and a tram. In the background, there are other cars moving slowly and changing of direction. There are many perturbations in this sequence, such as noise or changes in the illumination that have influence in the optical flow estimation.

[4] http://sintel.is.tue.mpg.de/

[5] `http://i21www.ira.uka.de/image_sequences`

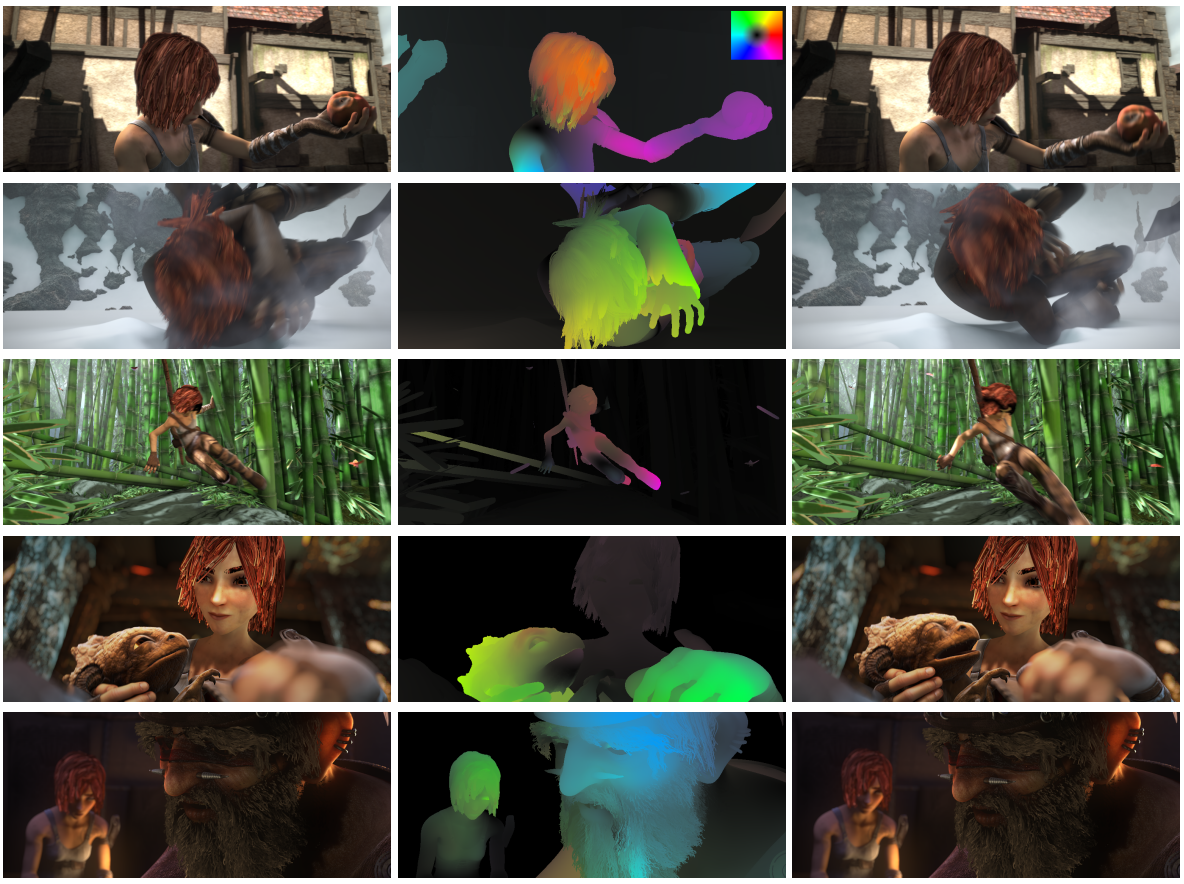


Figure 2.8: MPI-Sintel Flow dataset. From top to bottom, the first and third columns show consecutive images from the sequences of *Alley 1*, *Ambush 4*, *Bamboo 2*, *Bandage 1* and *Shaman 2*, respectively. The central column depicts the corresponding true flows for every pair of images. The motion is represented with the color scheme showed in the first row.

Rheinhafen also belongs to the previous traffic scenes created by Nagel[6]. It is a grayscale sequence composed by 1000 frames with a size of 688×565 pixels. The scene is made with a camera located at a certain height capturing the road traffic that circulates in one way. Several vehicles are seen circulating in different directions and speeds. Next to the camera you will catch a van that circulates at high speed. In the background, you see several vehicles that move at a slower speed and that are arranged to change direction. In this sequence the effect of *interlacing* is clearly perceived in the contours of the van near the chamber.

The sequence of *Ettlinger-Tor* is traffic scene of grayscale images (512×512) that shows an intersection recorded at the Ettlinger-Tor in Karlene by a stationary camera.

[6] `http://i21www.ira.uka.de/image_sequences`

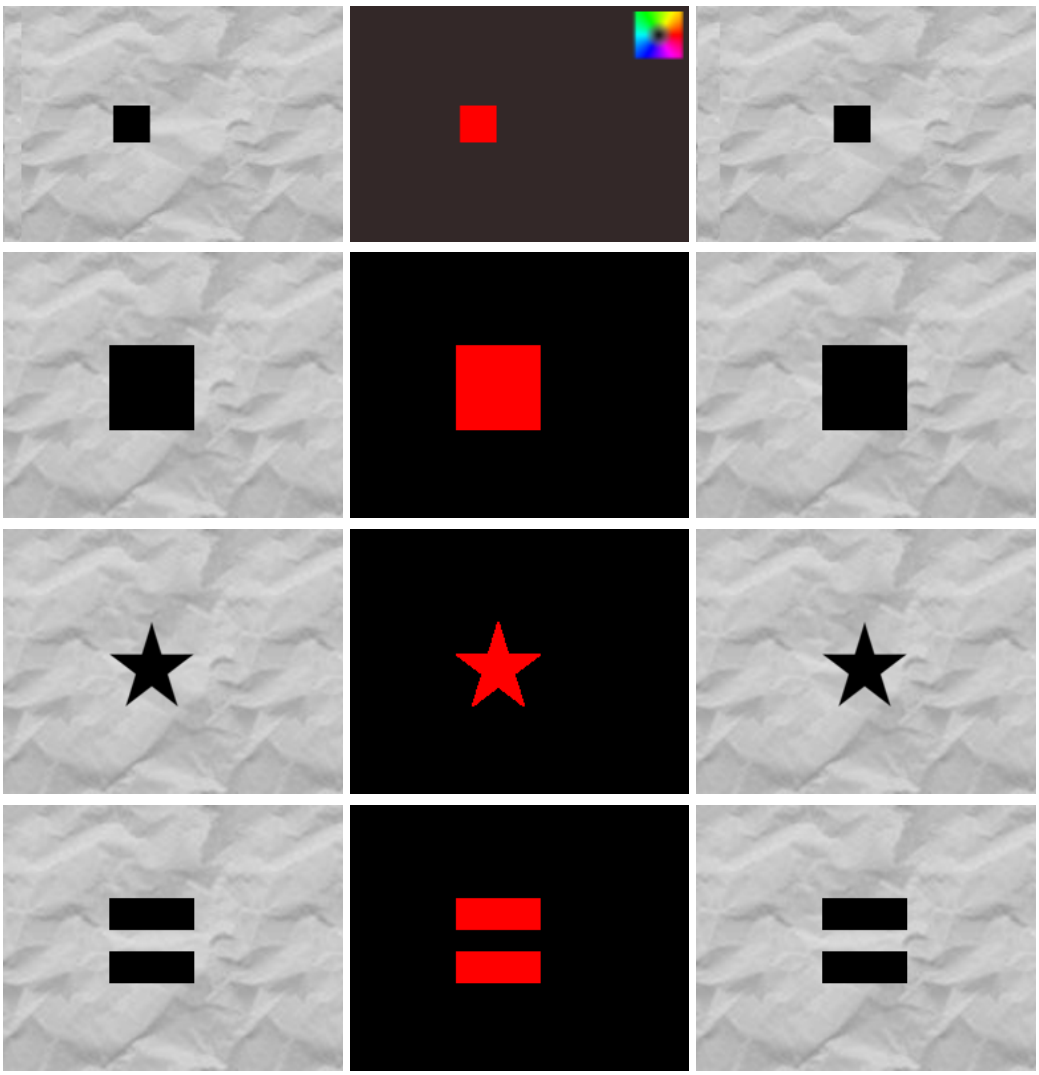


Figure 2.9: Several sequences of the *Geometric* dataset developed by the AIRS group. The motion is represented with the color scheme showed in the first row.

Finally, the sequences of *Moving Arm*, *GULP-Car* and *Taxi* have been recorded by the AIRS group. In the first one, a member of this group appears displacing radially his right arm. It is a sequence with several challenges like noise and undesired motion of the camera. The second scene shows a car leaving the computer engineering building of the GULP. In the third sequence we observe a moving taxi in front of the previous building.

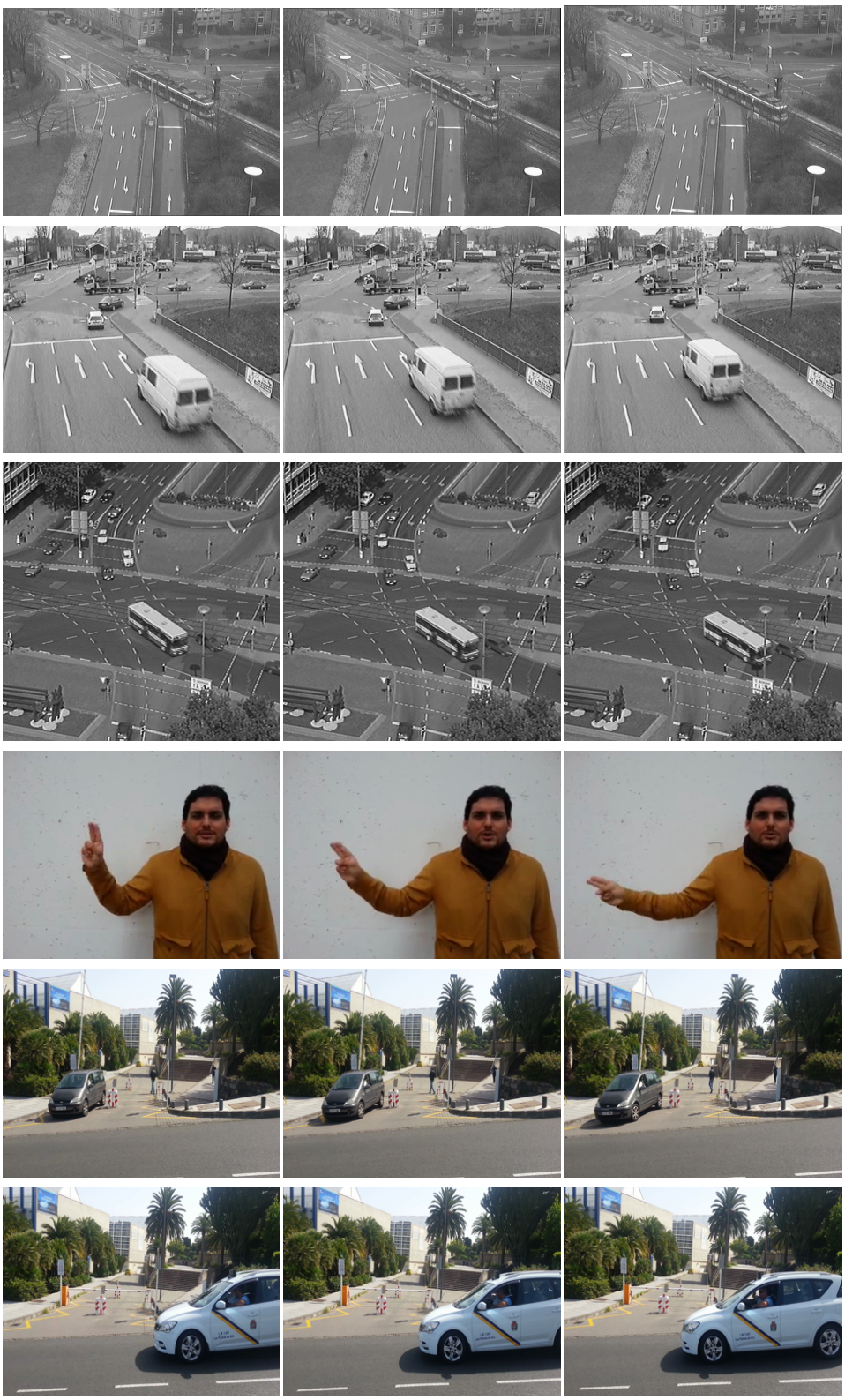


Figure 2.10: From top to bottom: Several frames from the realistic sequences of *Karl-Wilhelm-strasse*, *Rheinhafen*, *Ettlinger-Tor*, *MovingArm*, *ULPGC-Car* and *Taxi*.

2.7 Reproducible Research

The articles of Sánchez *et al.*[7] [Sánchez13b] and Monzón *et al.*[8] [Monzón16b] were published in the Image Processing On Line (IPOL) journal[9], which logo can be seen in figure 2.11. This is a scientific journal on mathematical signal processing algorithms (image, video, audio, 3D) which accentuates on the importance of reproducibility.

Figure 2.11: IPOL logo. Journal for reproducible research.

The objectives of its editorial committee differ from other classic journals: each IPOL article must present a complete description of its mathematical details together with a precise explanation of its methods with pseudo-codes. These ones must describe exactly the implementation that achieves the results depicted in the paper. The idea is that readers with sufficient skills could implement their own version (in any programming language or environment) directly from the IPOL article. Furthermore, submitting an IPOL paper means to upload the manuscript and the original source codes. Here, the role of referees includes a depth revision of the quality of the publication and that the pseudo-codes match exactly with the attached program, before the editor decision. The publication process is divided in two stages: first, the reviewers evaluate the scientific interest, the experiments and the reproducibility of the work; secondly, if this evaluation is positive, the authors submit the original code and the on-line demo is published.

This mechanism confronts the traditional publishing system where the method and some details about the implementation are usually described but, it is not always possible to confirm accurately the published results. Normally, we need to download the source code (if its available) and make it work in our computers to reproduce the results. This is not always immediate because the compilation of a code, its installation and even its use is not necessarily easy. In fact, it is not strange that several problems may appear during the installation process before we can execute our first test. In this sense, the work flow of an IPOL demo changes the traditional steps allowing to test the method even before considering if a researcher is going to invest more time in study a research work in depth. Regarding to this, the demo only executes the algorithms provided by the authors. In this sense, each IPOL demo downloads and compiles by itself the source code. This ensures that the users can reproduce exactly the results claimed by the authors.

[7] http://www.ipol.im/pub/art/2013/21/
[8] http://www.ipol.im/pub/art/2016/172/
[9] http://www.ipol.im/

The demo can also archive the experiments made with freely uploaded input data. This facilitates that distant researchers can collaborate remotely and that external users experiment with the demo using their own images. In this regard, a method can extend its original purposes beyond the initial ideas of its authors. Besides, the number experiments made with the archive are stored since its first use. Thus, the amount of archived data reveals if the research community uses (or not) the proposals of an IPOL work beyond of the citations of the paper. Since it was started in 2010 and according the usage statistics collected along these years, IPOL has about 250 unique visitors per day.

The IPOL intention is to create an environment for the evaluation, preservation and diffusion of the image science due to its reproducible research, executable algorithms and experiment sharing. All IPOL software must be published with an open source license for normal verification and reuse of the code by researchers. This is mandatory for reproducible research. In the fourth appendix of this document, we present a detailed explanation of the work flow and the possibilities of the IPOL work, Monzón *et al.* More details about the IPOL project can be seen in the research works [Limare11, Colom15] and, extensively, in the thesis of Nicolas Limare [Limare12].

2.8 Error Measures

We evaluate the quality of the methods described in this dissertation by using standard error metrics. These ones allow to quantify the error between the perfect solutions (true flow) and our estimations.

In [Baker07a] was exposed a dataset that updates the reference sequences of their time and four error metrics. They are known as *Average Angular Error* (AAE), eq. (2.15), *End-point error* (EPE), eq. (2.16), *Interpolation error* and *Normalized Interpolation error*. These last two measure the precision of the interpolation in not dense solutions. We do not use them because our methods produce dense optical flows and, actually, the metrics most commonly used in the optical flow literature are the AAE and the EPE; thus, we use them in our work. The AAE is found by normalizing the vectors, taking the dot product, and calculating the inverse cosine of their dot product while the EPE is the euclidean distance between our solution and the true flow.

$$AAE = \frac{180}{N\pi} \sum_{i=1}^{N} \arccos\left(\frac{\mathbf{w}_i \mathbf{w}_{trueFlow\ i} + 1}{\sqrt{|w_i|^2 + 1}\sqrt{|w_{trueFlow\ i}|^2 + 1}} \right) , \tag{2.15}$$

$$EPE = \frac{1}{N} \sum_{i=1}^{N} |\mathbf{w}_i - \mathbf{w}_{trueFlow\ i}| , \tag{2.16}$$

where N is the number of pixels in the image and $\mathbf{w}_i$ y $\mathbf{w}_{trueFlow\ i}$ are the corresponding

values of the displacement field in pixel i from the estimation and the true flow, respectively.

Chapter 3

Nonlinear Temporal Coherence in Optical Flow Methods

The seminal work of Horn and Schunck [Horn81] proposed a very basic idea on temporal coherence. They compute the flow field between the first two images and use this estimation as initial guess for the following frame. This is a simple predicting technique that introduces the idea of *temporal continuity* [Black94] of the calculations that should result in less computations. This idea is exploited later in several works [Weickert01, Brox04, Bruhn05b, Papenberg06] where the flow is predicted in the following time instant and updated with the observed data.

The most widely accepted case is a very simple formulation based on the regularization of the temporal derivative. This has shown to be effective when the motion field is continuous, as has been originally demonstrated in [Weickert01]. Although this is a simple idea, it is not realistic for many optical flow applications.

The main handicap of using the temporal derivative is the implicit assumption that the motion field must be continuous. This poses a sort of incongruency with respect to the nonlinear attachment of the data, which is specially designed for large displacements. When the temporal derivative is coupled with the spatial derivatives, it may also mitigate the effect of the spatial regularization.

In this chapter, we focus on improving the management of the temporal information in optical flow methods. The presence of large motions in natural scenes is very common, so it is necessary to investigate on more realistic temporal models. We propose a new way to handle the temporal information. In our opinion, this is more congruent than previous approaches. We claim that the separation between the spatial and temporal regularizations is necessary. The temporal formulations must also consider the non continuous nature of optical flow.

We propose several contributions: on the one hand, we introduce a nonlinear flow

constancy assumption (FCA) which is similar in spirit to the nonlinear data assumption; on the other hand, we propose a novel nonlinear flow regularization scheme (FRS) that can be seen as the counterpart of the temporal derivative for non continuous motions. We use a robust anisotropic diffusion operator based on the Nagel-Enkelmann operator. This operator allows respecting the object boundaries during the diffusion process, at the same time that it avoids oversegmentation in texture regions. This is similar to other approaches like, for instance, the ones presented in [Werlberger09] or [Zimmer11].

The former contribution is motivated by the results presented in [Salgado06], where a similar method was proposed based on the Nagel-Enkelmann diffusion operator. The numerical results showed that the use of a nonlinear temporal formulation of the flow field provided very good results. That was the first time that such a nonlinear flow assumption was introduced.

The second contribution introduces a non-continuous flow regularization scheme. In this case, we propose a new scheme, at the PDE level, which is similar in spirit to the temporal laplacian. In these two new assumptions, we devise the need to use the backward flow in order to find appropriate correspondences in previous frames. This is key to tracking the right motion of the objects in the sequence.

We estimate the backward correspondences once we have previously calculated the optical flow. In appendix I, we present four algorithms that use directly the forward optical flow and the image intensities for computing the backward flow. In our temporal method, we use the second algorithm due to its good results.

In order to recover large displacements, we implement a coarse-to-fine strategy. The optical flow is initialized at the coarsest scale and incrementally refined in the following scales. This is a common approach to deal with large displacements in variational optical flow methods. Another approach, which is very similar in spirit to this technique, is the use of a linear scale-space focusing strategy, as presented in [Alvarez00].

In the experiments, we observe that the temporal dimension allow finding continuous flow fields in time and the noise is considerably reduced with respect to the solution given by the spatial method.

The chapter is organized as follows: In section 3.1, we examine the new energy model and explain the novel temporal coherence strategy; the energy model is minimized in section 3.1.1 and some numerical details are explained. The new nonlinear flow smoothing scheme is joined to the resulting PDE in section 3.1.1. In the experimental results, we test our method using the *Middlebury* benchmark datasets, a geometric sequence and with realistic scenes. We also present the results provided by the method compared with other strategies presented in *Middlebury*. Finally, section 3.4 shows the conclusions of our work.

3.1 Energy Functional

Let $I_j(\mathbf{x})$ be a set of images, with $j = 1, .., N$, N the number of frames and $\mathbf{x} = (x, y)$. The optical flows are defined as $\{\mathbf{w}_i(\mathbf{x})\}$, with $i = 1, .., N-1$. In the method, the energy functional is decomposed in two parts,

$$E\left(\{\mathbf{w}_i\}\right) = E_S\left(\{\mathbf{w}_i\}\right) + E_T\left(\{\mathbf{w}_i\}\right), \tag{3.1}$$

where E_S, is the spatial energy model and E_T stands for the energy model corresponding to the temporal coherence strategy.

The spatial model, E_S, includes an attachment term based on brightness and gradient constancy assumptions and an anisotropic smoothness term. E_S accounts for the estimation of motion fields between two consecutive frames and reads as follows:

$$\begin{aligned} E_S = & \int \sum_{i=1}^{N-1} \Psi\left(\left(I_i(\mathbf{x}) - I_{i+1}(\mathbf{x}+\mathbf{w}_i(\mathbf{x}))\right)^2\right) \mathbf{dx} \\ & + \gamma \int \sum_{i=1}^{N-1} \Psi\left(\left\|\nabla I_i(\mathbf{x}) - \nabla I_{i+1}(\mathbf{x}+\mathbf{w}_i(\mathbf{x}))\right\|^2\right) \mathbf{dx} \\ & + \alpha \int \sum_{i=1}^{N-1} \Psi\left(\mathcal{N}\left(\nabla I_i, \nabla \mathbf{w}_i\right)\right) \mathbf{dx}, \end{aligned} \tag{3.2}$$

with $\Psi\left(s^2\right) = \sqrt{s^2 + \epsilon^2}$ (ϵ a prefixed small constant, e.g. 0.01). We use the anisotropic diffusion operator, $\mathcal{N}\left(\nabla I_i, \nabla \mathbf{w}_i\right) = trace\left(\nabla \mathbf{w}_i^T(\mathbf{x}) D(\nabla I_i) \nabla \mathbf{w}_i(\mathbf{x})\right)$, proposed in [Nagel86]. It combines the benefits of a TV-L^1 approach with the anisotropic behavior of this operator. $D(.)$ is defined as:

$$D(\nabla I) = \frac{\nabla I^{\perp} \nabla I^{\perp T} + \lambda^2 \mathbf{Id}}{\left\|\nabla I\right\|^2 + 2\lambda^2},$$

with **Id** the identity matrix. λ determines the gradient value from which the anisotropy is activated.

Introducing the *Flow Constancy Assumption* (FCA)

Using the temporal derivative for estimating the optical flow means that the motion of the objects need to be continuous in time or, equivalently, that the objects move slowly. Following the work [Salgado06], we avoid calculating temporal derivatives. Given that an object in the sequence may undergo large displacements we have to deal with information

that is warped through the flows. In fact, given a flow $\mathbf{w}_i(\mathbf{x})$, at instant i, its corresponding flow at the following time instant is $\mathbf{w}_{i+1}(\mathbf{x}+\mathbf{w}_i(\mathbf{x}))$. If $\mathbf{w}_i(\mathbf{x})$ is big then the temporal derivative is no longer valid, but the previous correspondence still holds. The next step is to pose any smoothness or attachment constraint that could relate these correspondences. One direct is what we call the *flow constancy assumption*, FCA, that is:

$$\mathbf{w}_i(\mathbf{x}) = \mathbf{w}_{i+1}(\mathbf{x}+\mathbf{w}_i(\mathbf{x})).$$

This formulation is similar in spirit to the traditional Lambertian assumption for large displacements in the image intensities. As we could expect, this type of assumptions should favor a constancy of the flow so it is well suited for detecting translational displacements with constant velocities. This lead us to propose a new kind of *flow attachment* terms in the energy model:

$$\Phi\left(\|\mathbf{w}_i(\mathbf{x}) - \mathbf{w}_{i+1}(\mathbf{x}+\mathbf{w}_i(\mathbf{x}))\|^2\right), \tag{3.3}$$

with $\Phi(.)$ any of the traditional quadratic or robust error function in the literature.

On the other hand, we devise in our energy functional (eq. (3.1)) a temporal coherence model, E_T, based on a nonlinear formulation. This temporal model favors the continuity of the flow fields in time.

$$E_T = \beta \int \sum_{i=1}^{N-2} \Phi\left(\|\mathbf{w}_i(\mathbf{x}) - \mathbf{w}_{i+1}(\mathbf{x}+\mathbf{w}_i(\mathbf{x}))\|^2\right) \mathbf{dx}, \tag{3.4}$$

with $\Phi\left(s^2\right) = e^{-\|\nabla I\|^\kappa}\sqrt{s^2+\epsilon^2}$, with $\kappa = 0.8$ and $\epsilon = 0.01$. This function has been used in recent spatial methods, like in [Xu10], [Wedel09a], to make coincident the contours of the motion field with the contours of the objects. In our case, it is useful to penalize the temporal attachment at the image discontinuities, in order to reduce the effect of occlusion propagation in time. The behavior of exponential functions in the regularization term will be deeply described in chapter 5 but in a spatial model.

This temporal energy (3.4) can be regarded as an attachment of the flow field in time. Its formulation is congruent with the brightness and gradient constancy terms. In the presence of large displacements, this temporal model is coherent with the spatial formulation and relates values at the correct positions. Note that when object displacements are very small, this term can be seen as an approximation of a continuous temporal regularization scheme based on the flow temporal derivative, which has been previously used in several works (e.g., [Weickert01]). On the contrary, the temporal derivative is useless in the presence of large displacements.

Predictably, the attachment of the flow will benefit the detection of constant motions in the sequence. This is not the most general displacements that may appear in a sequence,

but thanks to the robust function, it may also approximate small inertial motions. Later, in section 3.1.1 we will see that this term approximates a nonlinear smoothing scheme. Also, in section 3.1.1 we introduce a pure nonlinear temporal smoothing approach, which is similar to the minimization of the temporal derivative. That kind of terms is more suitable for general inertial motions.

3.1.1 Minimizing the Energy Functional

In this section, we derive the Euler-Lagrange of equation (3.1) into a nonlinear regularization scheme which has the same effect as the Laplacian for the continuous case.

The Euler-Lagrange equations for the spatial energy model (3.2) is:

$$\begin{aligned}\mathbf{0} &= \Psi'\left(\left(I_i(\mathbf{x}) - I_{i+1}(\mathbf{x}+\mathbf{w}_i(\mathbf{x}))\right)^2\right)\\ &\quad\cdot\left(I_i(\mathbf{x}) - I_{i+1}(\mathbf{x}+\mathbf{w}_i(\mathbf{x}))\right)\cdot\nabla I_{i+1}(\mathbf{x}+\mathbf{w}_i(\mathbf{x}))\\ &\quad+\gamma\ \Psi'\left(\left\|\nabla I_i(\mathbf{x}) - \nabla I_{i+1}(\mathbf{x}+\mathbf{w}_i(\mathbf{x}))\right\|^2\right)\\ &\quad\cdot\left(\nabla I_i(\mathbf{x}) - \nabla I_{i+1}(\mathbf{x}+\mathbf{w}_i(\mathbf{x}))\right)\cdot\mathcal{H}I_{i+1}(\mathbf{x}+\mathbf{w}_i(\mathbf{x}))\\ &\quad+\alpha\ \mathrm{div}\left(\Psi'\left(\mathcal{N}\left(\nabla I_i, \nabla\mathbf{w}_i\right)\right)\cdot D\left(\nabla I_i\right)\cdot\nabla\mathbf{w}_i\right),\end{aligned} \tag{3.5}$$

where $\mathcal{H}I_{i+1}$ is the Hessian matrix.

In order to derive $(\mathbf{w}_i(\mathbf{x}) - \mathbf{w}_{i+1}(\mathbf{x}+\mathbf{w}_i(\mathbf{x})))$ with respect to $\mathbf{w}_{i+1}(\mathbf{x})$, it is necessary to realize a change of variables, $\mathbf{z} = \mathbf{x} + \mathbf{w}_{i-1}(\mathbf{x})$. This change allows us to remove the nonlinearity inside the flow. The backward flow, $\mathbf{w}^*_{i-1}$, appears due to this change of variables. These ideas were drawn in [Salgado06] but, in that case, the backward flow was directly introduced in the energy model. We compute the backward flow directly from the forward flow, avoiding the need to introduce a symmetric model.

The minimization of the temporal coherence term is not evident because of the nonlinearity in the flow functions. We show that in this minimization the backward flow naturally appears, which is an interesting issue in the temporal diffusion process. This enables a temporal smoothing that transmits the information between the past and future estimates.

Starting with the temporal energy term

$$J(\{\mathbf{w}_i\}) = \int\sum_{i=1}^{N-2}\Phi\left(\left\|\mathbf{w}_i(\mathbf{x}) - \mathbf{w}_{i+1}(\mathbf{x}+\mathbf{w}_i(\mathbf{x}))\right\|^2\right)\mathbf{dx}, \tag{3.6}$$

with $\mathbf{w}_i(\mathbf{x}) = (u_i(\mathbf{x}), v_i(\mathbf{x}))^T$, the problem arises due to the composition of $\mathbf{w}_{i+1}$ with $\mathbf{w}_i$. Before the minimization, it is necessary to make a change of variables in the integral to deal with this nonlinearity.

Expanding the sum in (3.6), the temporal energy functional reads as

$$\begin{aligned} J(\{\mathbf{w}_i\}) = &\int \Phi\left(\|\mathbf{w}_1(\mathbf{x}) - \mathbf{w}_2(\mathbf{x}+\mathbf{w}_1(\mathbf{x}))\|^2\right)\mathbf{dx} \\ &+ \int \Phi\left(\|\mathbf{w}_2(\mathbf{x}) - \mathbf{w}_3(\mathbf{x}+\mathbf{w}_2(\mathbf{x}))\|^2\right)\mathbf{dx} \\ &+ \dots \\ &+ \int \Phi\left(\|\mathbf{w}_{N-2}(\mathbf{x}) - \mathbf{w}_{N-1}(\mathbf{x}+\mathbf{w}_{N-2}(\mathbf{x}))\|^2\right)\mathbf{dx}. \end{aligned} \tag{3.7}$$

We observe that every flow function appears in two terms, except for the first and last flow fields, $\mathbf{w}_1$ and $\mathbf{w}_{N-1}$. First analyzing the interior terms, with $i = 2, \dots, N-2$, we have two components for each $\mathbf{w}_i$,

$$\begin{aligned} J(\|\mathbf{w}_i\|) =& J(\|\mathbf{w}_i\|)_1 + J(\|\mathbf{w}_i\|)_2 \\ =& \int \Phi\left(\|\mathbf{w}_{i-1}(\mathbf{x}) - \mathbf{w}_i(\mathbf{x}+\mathbf{w}_{i-1}(\mathbf{x}))\|^2\right)\mathbf{dx} \\ &+ \int \Phi\left(\|\mathbf{w}_i(\mathbf{x}) - \mathbf{w}_{i+1}(\mathbf{x}+\mathbf{w}_i(\mathbf{x}))\|^2\right)\mathbf{dx}. \end{aligned} \tag{3.8}$$

The problem with the first component, $J(\{\mathbf{w}_i\})_1$, is that $\mathbf{w}_i$ depends on $(\mathbf{x}+\mathbf{w}_{i-1}(\mathbf{x}))$ and it is not possible to minimize it directly. We may proceed with the following change of variables: $\mathbf{z} = \mathbf{x} + \mathbf{w}_{i-1}(\mathbf{x})$, so that $\mathbf{w}_i(\mathbf{x}+\mathbf{w}_{i-1}(\mathbf{x}))$ boils down to $\mathbf{w}_i(\mathbf{z})$. Then, we can isolate the variable $\mathbf{x} = \mathbf{z} + \mathbf{w}^*_{i-1}(\mathbf{z})$, with $\mathbf{w}^*_{i-1}(\mathbf{z}) = \left(u^*_{i-1}(\mathbf{z}), v^*_{i-1}(\mathbf{z})\right)^T$, if it fulfills $\mathbf{w}^*_{i-1}(\mathbf{z}) + \mathbf{w}_{i-1}(\mathbf{z} + \mathbf{w}^*_{i-1}(\mathbf{z})) = 0$. This means that $\mathbf{w}^*_{i-1}(\mathbf{z})$ must hold the following equality:

$$\mathbf{w}^*_{i-1}(\mathbf{z}) = -\mathbf{w}_{i-1}(\mathbf{z} + \mathbf{w}^*_{i-1}(\mathbf{z})). \tag{3.9}$$

In this sense, if $\mathbf{w}_{i-1}$ is the flow field from frame $i-1$ to frame i, then $\mathbf{w}^*_{i-1}$ can be interpreted as the flow field from frame i to $i-1$, i.e., the *backward flow*. This requires that $\mathbf{w}_{i-1}$ be injective. In general, this is true in the whole domain, except for occluded regions, where several points of the domain may have the same target. Thus, occlusions should have a different treatment as it is explained in appendix I.

The Jacobian matrix for this change of variables is:

$$\begin{aligned}\mathcal{J}(\mathbf{z}) = &\begin{vmatrix} 1+u^*_{i-1_x}(\mathbf{z}) & u^*_{i-1_y}(\mathbf{z}) \\ v^*_{i-1_x}(\mathbf{z}) & 1+v^*_{i-1_y}(\mathbf{z}) \end{vmatrix} \\ &= \left(1+u^*_{i-1_x}(\mathbf{z})\right)\left(1+v^*_{i-1_y}(\mathbf{z})\right) \\ &\quad - u^*_{i-1_y}(\mathbf{z})v^*_{i-1_x}(\mathbf{z}), \end{aligned} \tag{3.10}$$

so the integral now becomes

$$\begin{aligned}J(\|\mathbf{w}_i\|) = \int \Phi\left(\left\|\mathbf{w}_{i-1}(\mathbf{z}+\mathbf{w}^*_{i-1}(\mathbf{z})) - \mathbf{w}_i(\mathbf{z})\right\|^2\right) \\ \cdot |\mathcal{J}(\mathbf{z})| \cdot \mathbf{dz}.\end{aligned}$$

The Euler-Lagrange equation is now straight forward

$$\begin{aligned}\left(\frac{\partial J(\|\mathbf{w}_i\|)}{\partial \mathbf{w}_i}\right)_1 = &-2\Phi'\left(\left\|\mathbf{w}_{i-1}(\mathbf{z}+\mathbf{w}^*_{i-1}(\mathbf{z})) - \mathbf{w}_i(\mathbf{z})\right\|^2\right) \\ &\cdot \left(\mathbf{w}_{i-1}(\mathbf{z}+\mathbf{w}^*_{i-1}(\mathbf{z})) - \mathbf{w}_i(\mathbf{z})\right) \cdot |\mathcal{J}(\mathbf{z})|.\end{aligned}$$

On the other hand, the minimization of $J(\|\mathbf{w}_i\|)_2$ with respect to $\mathbf{w}_i(\mathbf{x})$, yields the following equation:

$$\begin{aligned}\left(\frac{\partial J(\|\mathbf{w}_i\|)}{\partial \mathbf{w}_i}\right)_2 =&2\Phi'\left(\|\mathbf{w}_i(\mathbf{x}) - \mathbf{w}_{i+1}(\mathbf{x}+\mathbf{w}_i(\mathbf{x}))\|^2\right) \\ &\cdot \left(\mathbf{w}_i(\mathbf{x}) - \mathbf{w}_{i+1}(\mathbf{x}+\mathbf{w}_i(\mathbf{x}))\right)^T \\ &\cdot \left(\mathbf{Id} - \nabla \mathbf{w}^T_{i+1}(\mathbf{x}+\mathbf{w}_i(\mathbf{x}))\right),\end{aligned}$$

where **Id** is the identity matrix. Putting both terms together, we finally obtain the minimization of (3.8) as

$$\begin{aligned}\frac{\partial J(\|\mathbf{w}_i\|)}{\partial \mathbf{w}_i} = &-2\Phi'\left(\left\|\mathbf{w}_{i-1}(\mathbf{x}+\mathbf{w}^*_{i-1}(\mathbf{x})) - \mathbf{w}_i(\mathbf{x})\right\|^2\right) \\ &\cdot \left(\mathbf{w}_{i-1}(\mathbf{x}+\mathbf{w}^*_{i-1}(\mathbf{x})) - \mathbf{w}_i(\mathbf{x})\right) \cdot |\mathcal{J}(\mathbf{x})| \\ &+ 2\Phi'\left(\|\mathbf{w}_i(\mathbf{x}) - \mathbf{w}_{i+1}(\mathbf{x}+\mathbf{w}_i(\mathbf{x}))\|^2\right) \\ &\cdot \left(\mathbf{w}_i(\mathbf{x}) - \mathbf{w}_{i+1}(\mathbf{x}+\mathbf{w}_i(\mathbf{x}))\right)^T \\ &\cdot \left(\mathbf{Id} - \nabla \mathbf{w}^T_{i+1}(\mathbf{x}+\mathbf{w}_i(\mathbf{x}))\right),\end{aligned} \tag{3.11}$$

with $i = 2, ..., N-2$.

For $i = 1$, the minimization only affects the first term of the sum, having the following expression:

$$\begin{aligned}\frac{\partial J(\|\mathbf{w}_i\|)}{\partial \mathbf{w}_1} =& 2\Phi'\left(\|\mathbf{w}_1(\mathbf{x}) - \mathbf{w}_2(\mathbf{x}+\mathbf{w}_i(\mathbf{x}))\|^2\right) \\ & \cdot \left(\mathbf{w}_1(\mathbf{x}) - \mathbf{w}_2(\mathbf{x}+\mathbf{w}_1(\mathbf{x}))\right)^T \\ & \cdot \left(\mathbf{Id} - \nabla \mathbf{w}_2^T(\mathbf{x}+\mathbf{w}_1(\mathbf{x}))\right). \end{aligned} \tag{3.12}$$

And, for $i = N-1$, the minimization only affects the last term, so we have

$$\begin{aligned}\frac{\partial J(\|\mathbf{w}_i\|)}{\partial \mathbf{w}_{N-1}} =& -2\Phi'\left(\left\|\mathbf{w}_{N-2}(\mathbf{x}+\mathbf{w}_{N-2}^*(\mathbf{x})) - \mathbf{w}_{N-1}(\mathbf{x})\right\|^2\right) \\ & \cdot \left(\mathbf{w}_{N-2}(\mathbf{x}+\mathbf{w}_{N-2}^*(\mathbf{x})) - \mathbf{w}_{N-1}(\mathbf{x})\right) \cdot |\mathcal{J}(\mathbf{x})|. \end{aligned} \tag{3.13}$$

Thus, equations (3.11), (3.12) and (3.13) are the Euler-Lagrange equations of (3.8). Then, the temporal energy model (3.4) yields the following Euler-Lagrange equations:

$$\begin{aligned}\mathbf{0} =& \beta\ \Phi'\left(\|\mathbf{w}_i(\mathbf{x}) - \mathbf{w}_{i+1}(\mathbf{x}+\mathbf{w}_i(\mathbf{x}))\|^2\right) \\ & \cdot \Big(\left(\mathbf{w}_i(\mathbf{x}) - \mathbf{w}_{i+1}(\mathbf{x}+\mathbf{w}_i(\mathbf{x}))\right)^T \\ & \cdot \left(\mathbf{Id} - \nabla \mathbf{w}_{i+1}^T(\mathbf{x}+\mathbf{w}_i(\mathbf{x}))\right)\Big) \\ & + \beta\ \Phi'\left(\left\|\mathbf{w}_i(\mathbf{x}) - \mathbf{w}_{i-1}(\mathbf{x}+\mathbf{w}_{i-1}^*(\mathbf{x}))\right\|^2\right) \\ & \cdot \left(\left(\mathbf{w}_i(\mathbf{x}) - \mathbf{w}_{i-1}(\mathbf{x}+\mathbf{w}_{i-1}^*(\mathbf{x}))\right) \cdot |\mathcal{J}(\mathbf{x})|\right), \end{aligned} \tag{3.14}$$

where $|\mathcal{J}(\mathbf{x})|$ stands for the absolute value of the determinant of the Jacobian matrix, which is equal to $\mathcal{J}(\mathbf{x}) = \left(1 + u_{i-1,x}^*\right)\left(1 + v_{i-1,y}^*\right) - u_{i-1,y}^* v_{i-1,x}^*$ with $\mathbf{w}_{i-1}^* = \left(u_{i-1}^*, v_{i-1}^*\right)^T$ as the backward flow from I_i to I_{i-1}.

We use a gradient descent approach to find the solution of the above PDE. The nonlinear terms are linearized using first order Taylor expansions. In the temporal dimension, we use Neumann boundary conditions for the last and first frames.

The first temporal term in (3.14) stands for the anticausal model, where only future and present information is used; and the second term accounts for the causal model, which depends on past, $\mathbf{w}_{i-1}$), and present data. Both parts conform a whole non-causal model. In the experimental results we examine these two terms separately, in order to study the effect of the causal and anticausal parts.

Introducing the nonlinear *Flow Regularization Scheme* (FRS)

In this section, we introduce a temporal smoothing term that aims at replacing the traditional continuous case, based on the minimization of $\frac{\partial \mathbf{w}_i(\mathbf{x})}{\partial t}$. The second order derivative of the flow field is usually approximated as $u_{tt} \approx u_{i,j,k+1} - 2u_{i,j,k} + u_{i,j,k-1}$

This has a temporal regularizing effect, centered at the same spatial position but at different time instants. It is clear that this scheme is valid if the flow field varies smoothly across the image sequence. This is not the case in the presence of large displacements.

We propose a new solution, which is similar in spirit to this numerical approximation, and is suitable to deal with non-continuous displacements. Following the reasoning of section 3.1, the corresponding flow positions at different time instants are located at the warping of the flow field. The counterpart of the continuous second order derivative to the nonlinear case, can be formulated as

$$\mathbf{w}_{i-1}(\mathbf{x} + \mathbf{w}^*_{i-1}(\mathbf{x})) - 2\mathbf{w}_i(\mathbf{x}) + \mathbf{w}_{i+1}(\mathbf{x} + \mathbf{w}_i(\mathbf{x})).$$

This is a nonlinear formulation that puts in correspondence the correct flow values of a given object in different frames. In the general case of inertial motions, this approximation can be considered as a nonlinear flow smoothing scheme, and it is the counterpart of the traditional continuous temporal regularization for non continuous motions. It is not evident how to abstract this term at the temporal energy. The use of robust functions can also be included as

$$\begin{aligned} T_S = \delta\ \Phi' \left(\left\| \mathbf{w}_{i-1}(\mathbf{x} + \mathbf{w}^*_{i-1}(\mathbf{x})) - \mathbf{w}_{i+1}(\mathbf{x} + \mathbf{w}_i(\mathbf{x})) \right\|^2 \right) \\ \cdot \left(\mathbf{w}_{i-1}(\mathbf{x} + \mathbf{w}^*_{i-1}(\mathbf{x})) - 2\mathbf{w}_i(\mathbf{x}) + \mathbf{w}_{i+1}(\mathbf{x} + \mathbf{w}_i(\mathbf{x})) \right). \end{aligned} \tag{3.15}$$

The robust function is the same as we have used so far, and it is based on the norm of the difference of the flow at the previous and following frames. This is very similar to the first order temporal derivative of the flow that is used in the continuous case. When the difference of the flow is big, it is treated as an outlier. δ is the weighting parameter and should be smaller than that of the spatial regularization.

This new term provides a new scheme at the PDE level that can be combined with the previous PDE equations (3.5) and (3.14). In the experiments we show that this new term provides good results. It has a similar gain as the continuous case, but it correctly handles the large displacements.

Also here, as in equation (3.14), it is easy to appreciate the contribution of the causal and anticausal parts.

3.2 Experimental Results

Next, we examine the behavior of the temporal models explained in this chapter. In the first row of figures 3.1 and 3.4, we show the third frames of the *small square* and *Urban2* sequences, respectively; their true flows, and the best spatial solutions found. In the second row, we show three temporal solutions: the first for the nonlinear temporal attachment defined in (FCA, eq. (3.4)); the second, for the nonlinear temporal smoothness approach defined in (FRS, eq. (3.15)); and, finally, using both temporal terms. In these experiments, we refer to this combination as Temporal Continuity Optical Flow (TCOF).

The *small square* is a simple sequence where the object is moved over a textured background. The square is moving 15 pixels per frame, while the background moves 3 pixels in the same direction. The most important features are its translational motion and a large displacement. On the other hand, we use the *Urban2* sequence from the *Middlebury* database due to its greater displacements (of about 20 pixels in some areas) respect to the other sequences of the dataset.

We show the AAE and EPE errors for every frames of the *small square*s in graphics of figures 3.3 and 3.2, respectively. In figure 3.5, we use *Urban2* to extend the comparison by showing motion details for the corresponding spatial and TCOF solutions regarding to its true flow. Then, we show the solutions for the spatial and TCOF approaches using the test and evaluation sequences from *Middlebury* in figures 3.6 and 3.7, respectively.

Finally, we use the sequence of *Karl-Wilhelm strasse* and a scene extracted from the movie *'Godfather: Part II'* in order to analyze the behavior of the proposed method in real world sequences. In figure 3.8, we depict the flow fields obtained with the spatial and spatio-temporal methods for several frames of the original sequences.

We represent the motion of these two last experiments with the IPOL color scheme while, the previous ones, are represented with the scheme of *Middlebury*.

According to the results, the improvement of the temporal methods with respect to the spatial solution is important. As expected, the spatial method produces higher errors at the motion discontinuities and, more significantly, at the occlusions. Table 3.1 shows the average End-point (EPE) and Angular (AAE) errors for the geometric sequence. The first temporal result, corresponding to the first image in the second row of figure 3.1, provides an important improvement on the EPE and, more noticeable, on the AAE. The improvement in accuracy is still more important if we use the nonlinear temporal smoothing scheme (3.15) or a combination of both.

We observe that the nonlinear temporal smoothing scheme (3.15) behaves better than the temporal attachment, even at the motion boundaries. The graphics in figure 3.2 show the EPE for every frame on the square sequence. Frame by frame, the optical flows are more accurate in the temporal methods. We also observe that the results are very stable, specially in the middle of the FRS line. Reasonably, the frames at the beginning and at

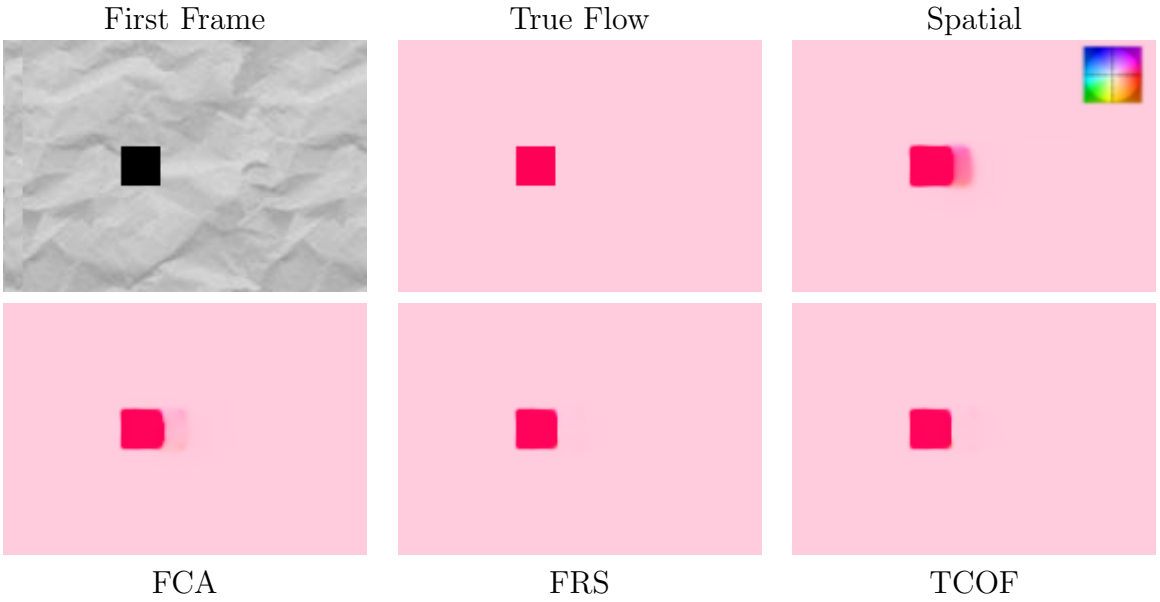


Figure 3.1: Square sequence. First row: one of the images of the Square sequence, the true flow and the best spatial solution found. Second row: three temporal solutions using FCA, FRS and TCOF approaches, respectively. The color scheme is showed in the upper-right corner of the spatial flow.

Table 3.1: EPE and AAE for the Square sequence.

Method	**EPE**	**AAE**
Spatial	0.071	0.629^o
FCA	0.049	0.204^o
FRS	0.036	$\mathbf{0.134}^o$
TCOF	**0.035**	0.138^o

the end of the sequence present higher errors, due to the Neumann boundary conditions. The AAE graphic in table 3.3 shows a similar behavior.

In the case of *Urban2*, the improvement of the temporal methods with respect to the spatial solution is not so evident. However, we observe better results dealing with the occlusions in the regions highlighted at figure 3.5.

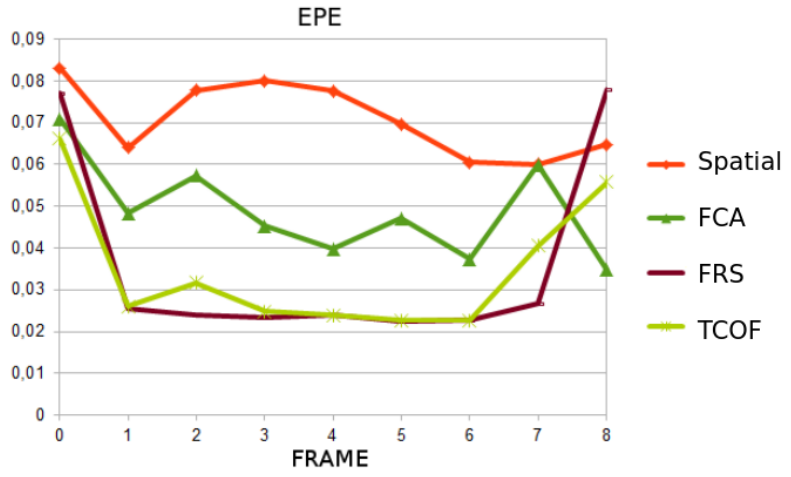


Figure 3.2: EPE in each optical flow of the Square sequence.

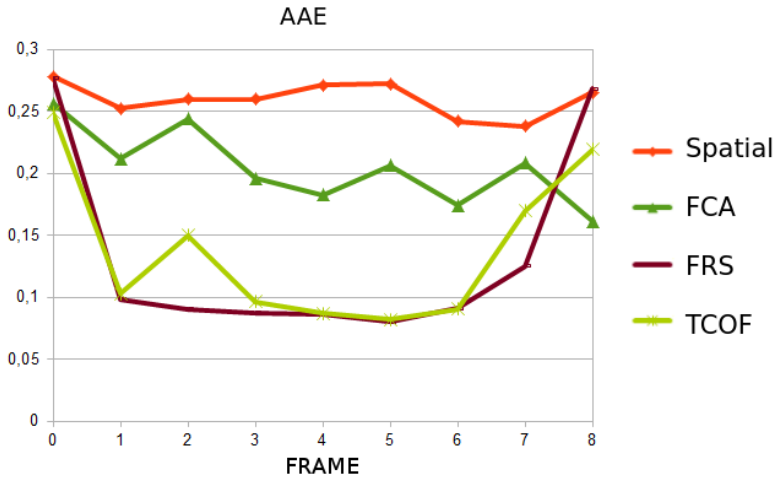


Figure 3.3: AAE in each optical flow of the Square sequence.

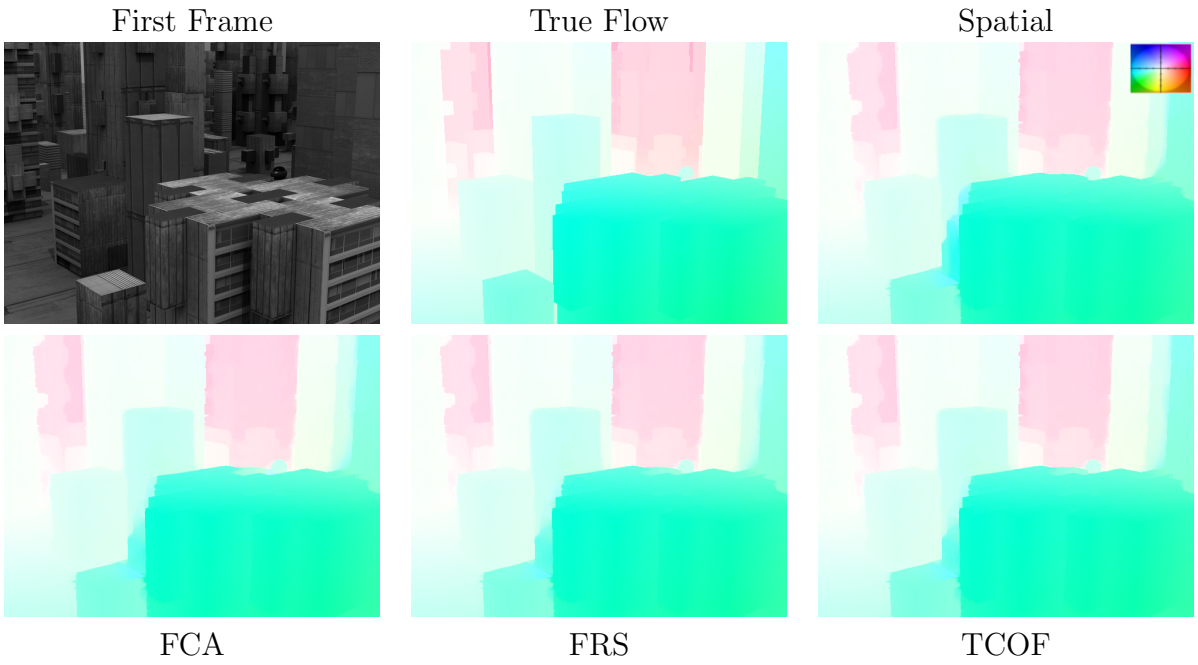


Figure 3.4: *Urban2*. First row: original frame, true flow and the best spatial solution found. Second row: Results for the three approximations that uses temporal coherence for the optical flow. The color scheme is showed in the upper-right corner of the spatial flow.

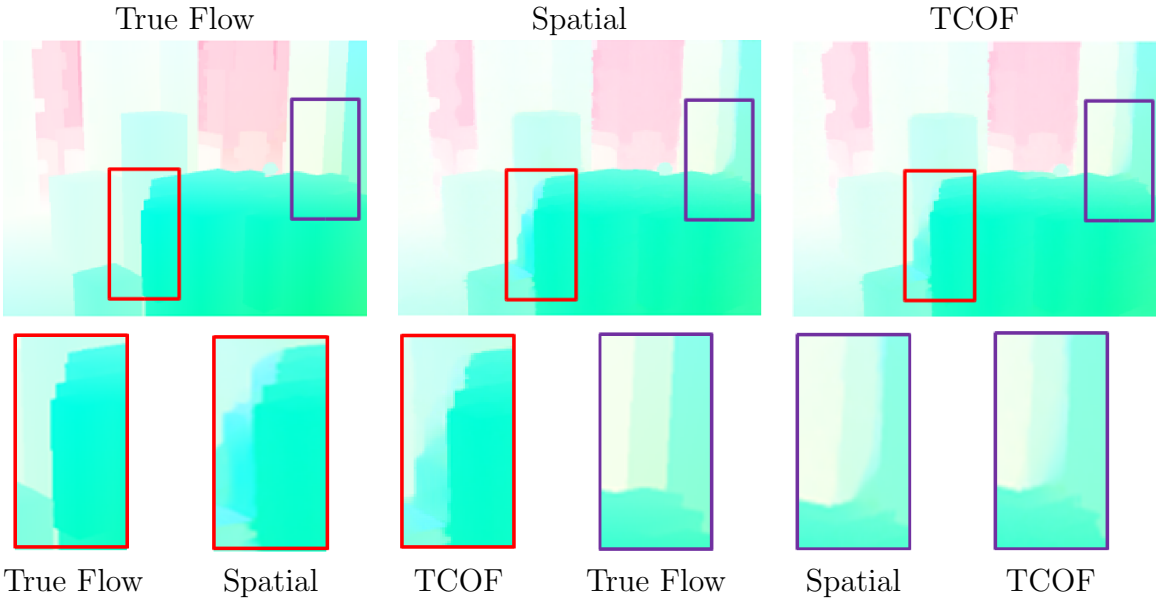


Figure 3.5: *Urban2* sequence. We show some flow details for comparing the spatial and temporal solutions regarding to the true flow.

Table 3.2: EPE and AAE for the *Middlebury* sequences.

Sequence	**Spatial EPE**	**Spatial AAE**	**TCOF EPE**	**TCOF AAE**
Hydrangea	0.165	2.086^o	0.162	2.032^o
Grove2	0.128	1.901^o	0.144	1.843^o
Grove3	0.559	5.781^o	0.552	5.682^o
Urban2	0.226	2.280^o	0.284	2.256^o
Urban3	0.338	2.633^o	0.317	2.429^o
RubberWhale	0.094	2.971^o	0.093	2.953^o

Next, we show the results for the test sequences of the *Middlebury* benchmark database [Baker07b], except Dimetrodon and Venus. This is because there are not enough images to use TCOF properly. The numerical results depicted in table 3.2 reflect that the temporal approach always improves the AAE respect to its spatial counterpart, specially in *Urban2* and *Urban3*. However, its EPE is worst in *Grove2*, *Urban2* and *RubberWhale* but not significantly. On the other hand, figure 3.6 depicts the original image, the true flow and the flow fields for the two approaches. Here, we observe accuracy in both solutions without strong differences. The TCOF method was sent to the *Middlebury* web page for its evaluation. Figure 3.7 contains the original image, the true flow and our solution. In general, we observe good results in the motion fields.

The optical flows obtained for the synthetic sequences of *Middlebury* do not show many differences between the spatial and temporal solution. This situation is the opposite in the real sequences depicted in figures 3.8 and 3.9. We observe that the temporal dimension allows finding continuous flow fields in time and the noise is reduced with respect to the solution given by the spatial method, specially in the *Karl-Wilhelm strasse* sequence. We also observe an improvement in the *Godfather* scene when the camera begin to move to the left. For instance, the details of the trees are better preserved in the temporal solution showed in the third row of figure 3.9. On the other hand, the first car that appears in the sequence leaves dust in the air when it gets out of the scene. When this happens, the temporal solution is more consistent than the spatial one. Both figures demonstrate that TCOF performance is better dealing with the complexity of real sequences.

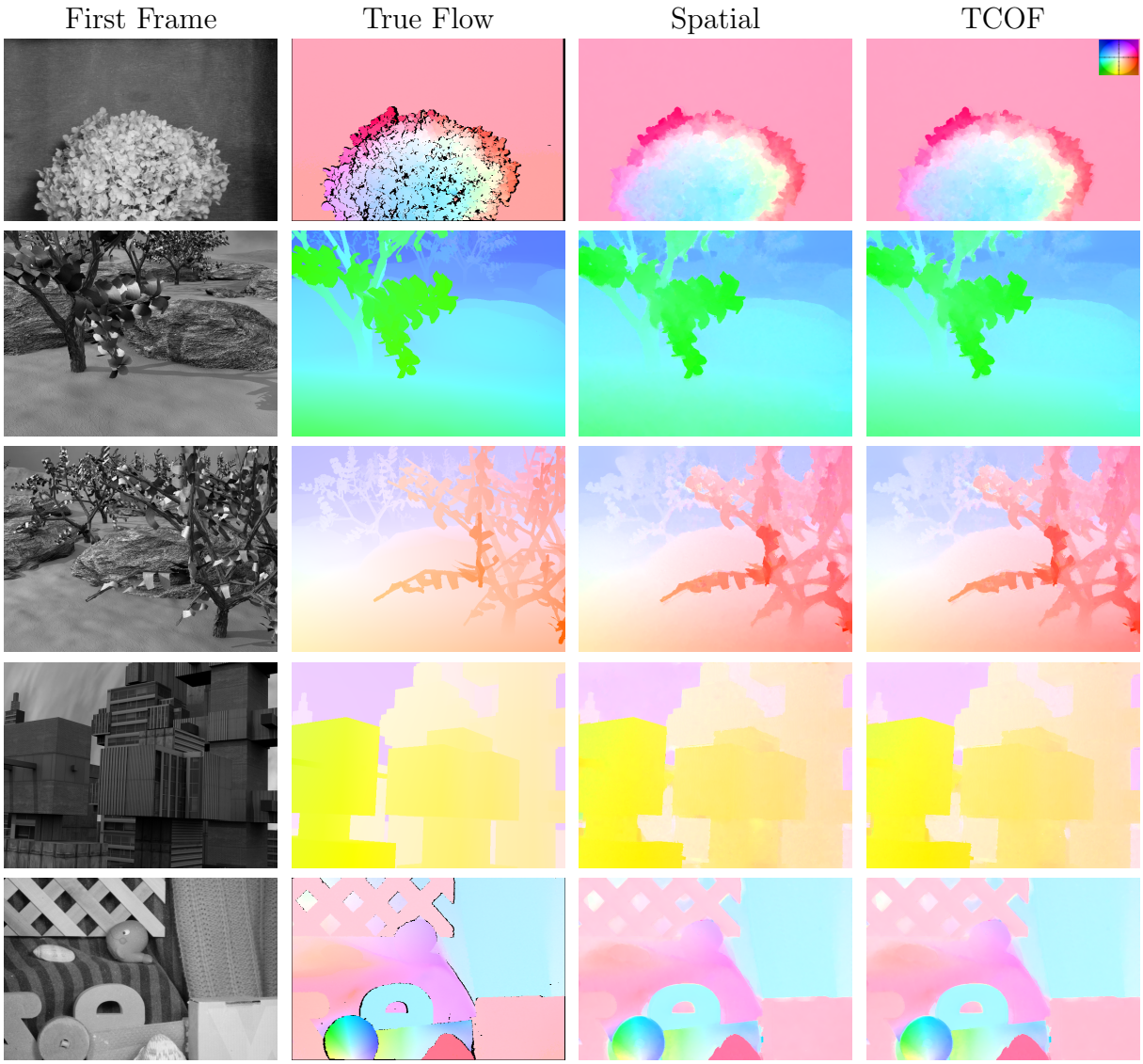


Figure 3.6: Optical flow fields using the temporal method on the test sequences of the *Middlebury* dataset. First row shows the original image, the second the true flow and the third the temporal solution. The color scheme is showed in the upper-right corner of the first temporal solution.

Figure 3.7: Optical flow fields using the temporal method on the evaluation sequences of the *Middlebury* dataset. First row shows the original image, the second the true flow and the third the temporal solution. The color scheme is showed in the upper-right corner of the first temporal solution.

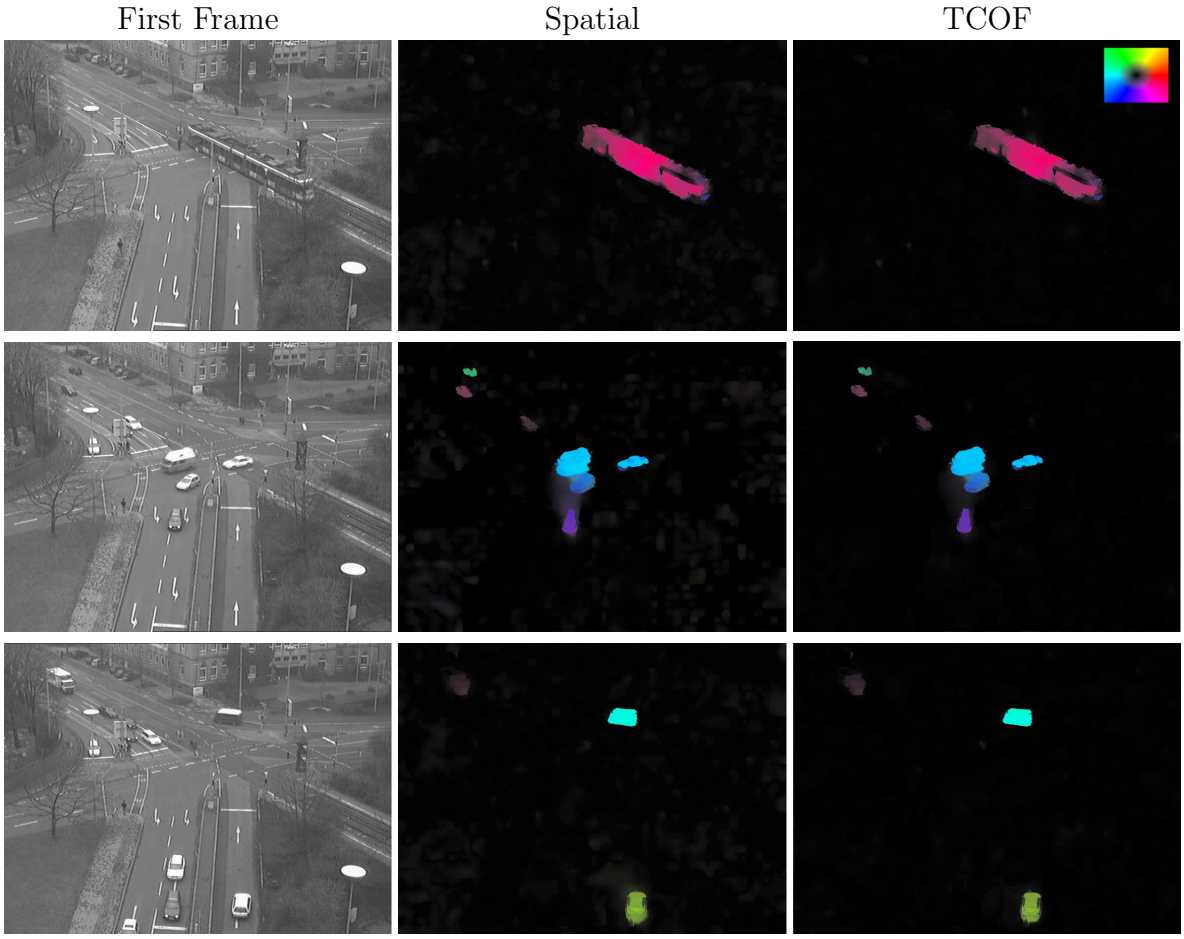


Figure 3.8: Comparison between the spatial and temporal solutions in the real sequence of *Karl-Wilhelm strasse*. The temporal coherence strongly reduces the noise regarding to the spatial counterpart. The color scheme is showed in the upper-right corner of the first temporal solution.

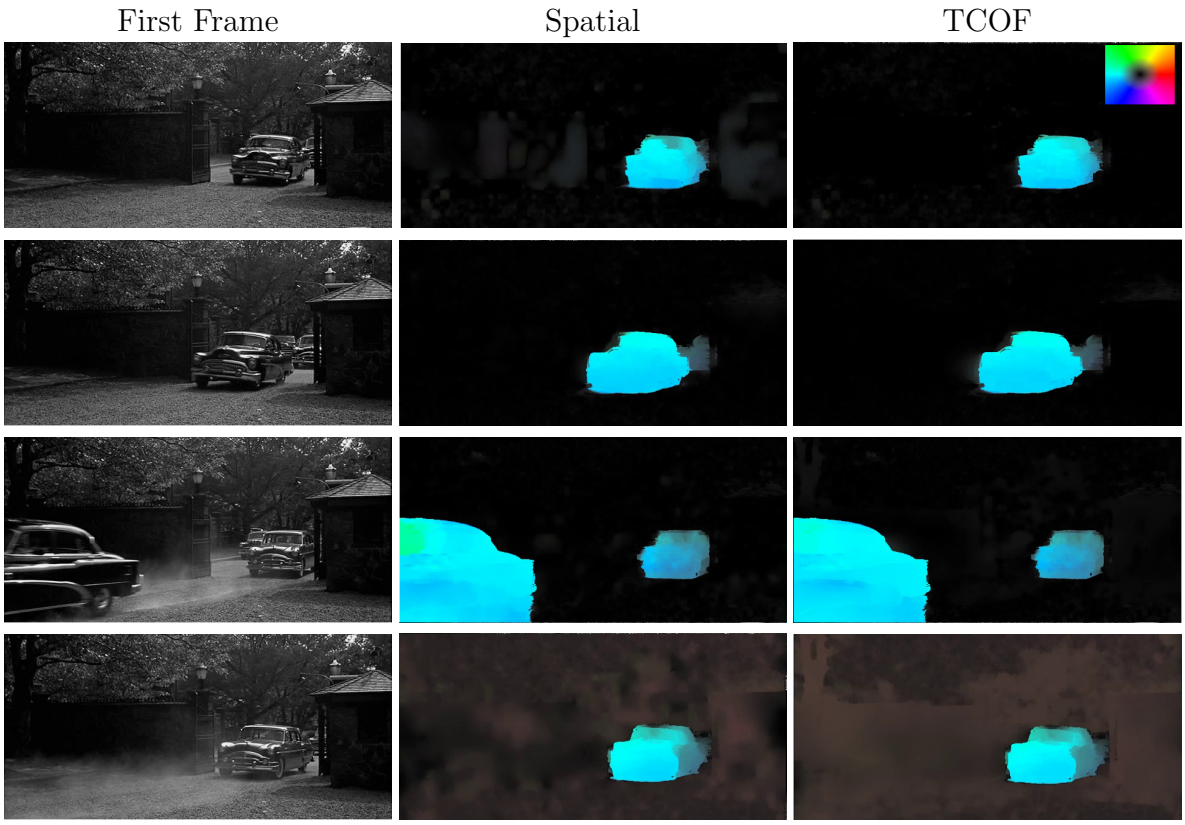


Figure 3.9: Comparison between the spatial and temporal solutions in a scene extracted from the *Godfather movie: Part II*. We observe an improvement due to the temporal information, specially in the first and third rows. The color scheme is showed in the upper-right corner of the first temporal solution.

3.3 Numerical Results

We extend our evaluation in tables 3.4 and 3.3, where we show the AAE and EPE achieved by TCOF, OFH [Zimmer11], TC-Flow [Volz11] and MDP-Flow2 [Xu12] approaches. We observe that the numerical differences are not pronounced except in the *Schefflera* sequence. Furthermore, we see better errors in the EPE of *Urban* and *Teddy*. This idea can be confirmed according to the flows depicted in figure 3.3, where can be seen that the differences are not very pronounced.

Table 3.3: EPE: *Middlebury* evaluation sequences.

Sequence	Rg. Eval.	TCOF	OFH [Zimmer11]	TC-Flow [Volz11]	MDP-Flow2 [Xu12]
Army	all	0.11	0.10	0.07	0.08
	disc	0.28	0.25	0.21	0.21
	untext	0.09	0.09	0.06	0.07
Mequon	all	0.24	0.19	0.15	0.15
	disc	0.76	0.69	0.59	0.48
	untext	0.19	0.14	0.11	0.11
Schefflera	all	0.53	0.43	0.31	0.20
	disc	1.15	1.02	0.78	0.40
	untext	0.29	0.17	0.14	0.14
Wooden	all	0.24	0.17	0.16	0.15
	disc	0.88	1.08	0.86	0.80
	untext	0.20	0.08	0.08	0.08
Grove	all	0.88	0.87	0.75	0.63
	disc	1.26	1.25	1.11	0.93
	untext	0.69	0.73	0.54	0.43
Urban	all	0.38	0.43	0.42	0.26
	disc	0.93	1.69	1.40	0.76
	untext	0.29	0.32	0.25	0.23
Yosemite	all	0.16	0.10	0.11	0.11
	disc	0.16	0.13	0.12	0.12
	untext	0.22	0.18	0.29	0.17
Teddy	all	0.49	0.59	0.62	0.38
	disc	1.03	1.40	1.35	0.79
	untext	0.65	0.74	0.93	0.44

Table 3.4: AAE: *Middlebury* evaluation sequences.

Sequence	Rg. Eval.	TCOF	OFH [Zimmer11]	TC-Flow [Volz11]	MDP-Flow2 [Xu12]
Army	all	4.17	3.90	2.91	3.23
	disc	10.4	9.77	8.00	7.93
	untext	3.71	3.62	2.34	2.60
Mequon	all	3.17	2.84	2.18	1.92
	disc	10.7	11.0	8.77	6.64
	untext	2.59	2.04	1.52	1.52
Schefflera	all	6.58	5.52	3.84	2.46
	disc	15.7	14.4	10.7	5.91
	untext	3.82	1.89	1.49	1.56
Wooden	all	3.69	3.52	3.13	3.05
	disc	16.1	20.5	16.6	15.8
	untext	2.37	1.60	1.46	1.51
Grove	all	3.78	3.18	2.78	2.77
	disc	4.95	4.06	3.73	3.50
	untext	2.47	2.82	1.96	2.16
Urban	all	2.59	3.86	3.08	2.86
	disc	8.47	14.1	11.4	8.58
	untext	2.58	3.59	2.66	2.70
Yosemite	all	3.66	1.77	1.94	2.00
	disc	4.83	3.62	3.43	3.50
	untext	2.67	1.81	3.20	1.59
Teddy	all	1.83	2.64	3.06	1.28
	disc	4.20	7.08	7.04	2.67
	untext	1.46	2.15	4.08	0.89

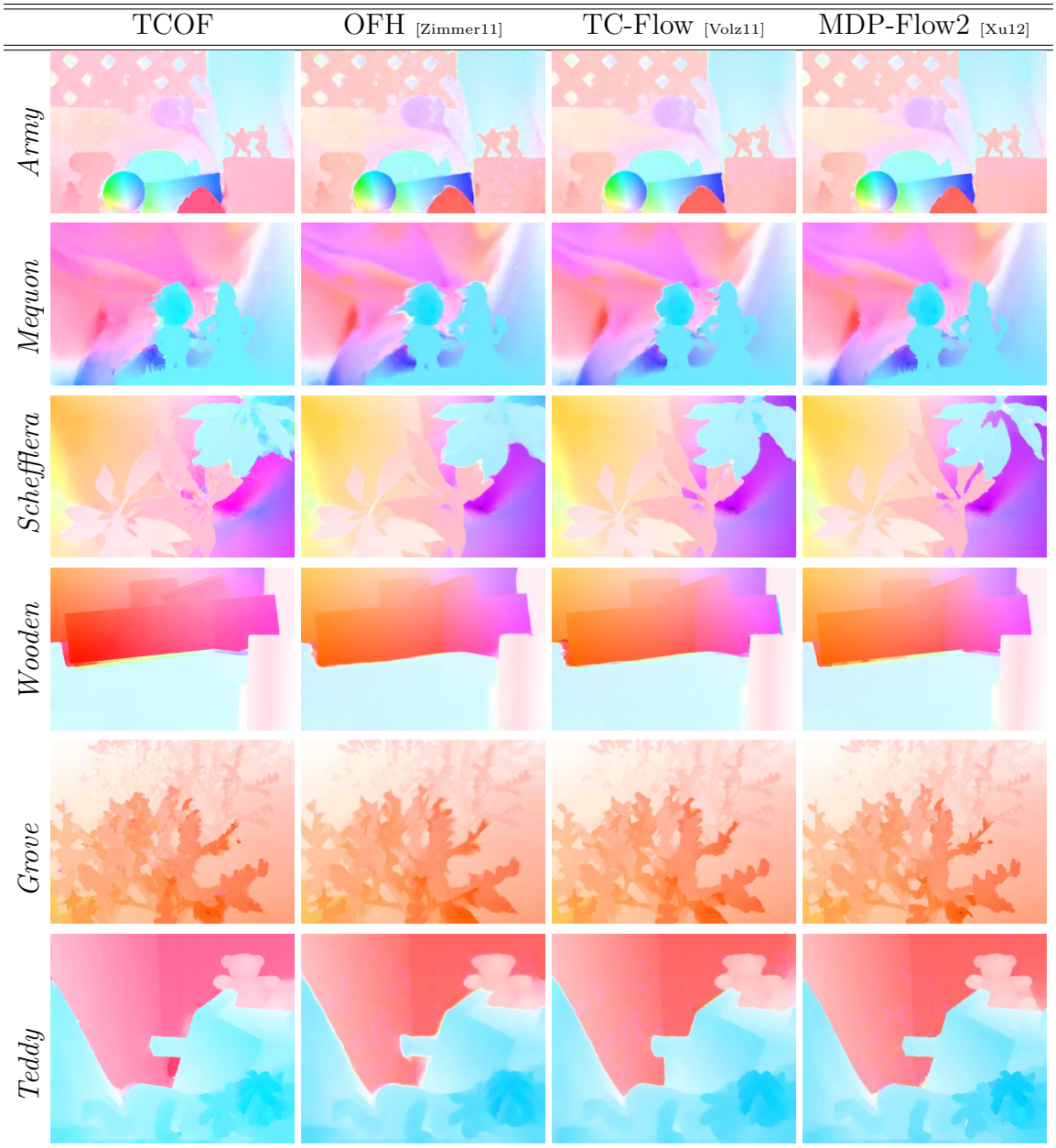


Figure 3.10: Flow fields obtained by the methods of TCOF, OFH, TC-Flow and MDP-Flow2 using the evaluation sequences from *Middlebury*.

3.4 Conclusion

In this chapter, we presented nonlinear flow assumptions for the estimation of motion fields with two contributions: on the one hand, we introduced a new flow constancy assumption (FCA) that is formulated as a non-linear attachment of the flow; on the other hand, a non-continuous regularization operator (FRS) at the PDE level, which regularizes the optical in the temporal dimension and is suitable for large displacements. These proposals are more confident in the estimation of motion fields than previous approaches.

We observe that the FCA term relates flow fields at different time instants and is consistent with the rest of the energy terms. In the experiments, we combine both ideas obtaining a scheme that conveniently deals with continuous and non-continuous velocities.

These nonlinear assumptions correctly fit with the standard nonlinear brightness and gradient constancy terms, can cope with general image sequences and provide better solutions. We observe that the TCOF scheme is more general than using the continuous temporal regularization of the flow, with the advantage that it conveniently deals with continuous and non-continuous velocities. In fact, if the motion is very small, this term approximates a continuous temporal smoothing scheme.

We have shown in the experimental results that using the backward flow is important to distribute good guesses all through the sequence of flows. The method provides important improvements, specially in the presence of large displacements. The results are promising in both cases, although we observe a better performance for the nonlinear temporal smoothing scheme in general. Another interesting result of the temporal coherence schemes is that the background motion oscillations tend to disappear. These oscillations clearly appear in the spatial method, in regions where there is no apparent motion.

Chapter 4

Robust Optical Flow Estimation

In this chapter, we analyze the optical flow method presented in Brox *et al.* [Brox04] using our own implementation[1]. First, we describe its Spatial and Temporal approaches for grayscale images. Both approximations are very similar but the Temporal approach includes a continuous smoothing scheme in the Temporal dimension. Second, we modified the Spatial approximation into a multi-channel extension to determine if the color information benefits the optical flow calculation.

This type of methods produce piecewise-smooth flow fields. In particular, Brox *et al.* proposed a technique that is more robust to outliers due to a continuous L^1 function that creates a TV regularization. It also deals with constant brightness variations by including a gradient constancy term in its energy functional in conjunction with the brightness constancy assumption [Horn81].

In our experiments, we first conduct a thorough analysis of the method with grayscale images from the *Middlebury* benchmark database. We observe that this kind of methods creates rounding shapes at the borders of the moving objects, which is a typical consequence of TV-L^1 strategies.

In relation to this, the color information could prevent this problem by associating the pixel information with various brightness intensities. Furthermore, this type of images may avoid the use of additional constraints in the flow calculation [Golland97, Ohta89] and contains more photometric information that can be useful against shadows, shading and image specularities [Barron02, vdW04, Mileva07]. On the other hand, we observe a better accuracy of the Spatial approach for the *Middlebury* sequences in our experiments. Thus, we introduce the color information in the Spatial version and compare with the basic approach.

Another alternative to avoid rounding shapes at the motion contours is to introduce a discontinuity preserving strategy in the regularization term similar as in [Wedel09a, Xu10].

[1] http://www.ipol.im/pub/art/2013/21/sms_optic_flow_2.0.tar.gz

In chapter 5, we combine both improvements and present several proposals into a multi-channel framework.

Once we conclude the experiments with grayscale images, we evaluate the possible benefits of color against grayscale information. We perform our evaluation with RGB and grayscale images in the sequences from the *Middlebury* benchmark database [Baker07b] and the open source movies from the *MPI-Sintel* flow data set [Butler12]. The purpose of these tests is to compare motion details between the best flows achieved for the same sequence. Our experiments demonstrate that, in general, the color information provides more accurate solutions and the computational cost seems reasonable given the improvement.

The chapter is organized as follows: first, Section 4.1 presents the grayscale optical flow model and the modification of its energy functional to introduce the general framework behind the multi-channel scheme; second, we deeply examine the behavior of the method for some standard image sequences and, then, we confront the color and grayscale information in the experiments of Section 4.2; next, we study the accuracy of both color spaces in Section 4.3; finally, a summary of the main ideas and conclusions are given in Section 4.4.

4.1 Energy Functional

We denote as $I : \Omega \subset \mathbb{R}^3 \to \mathbb{R}$ a sequence of grayscale images in space and time with $\mathbf{x} = (x, y, t)^T \in \Omega$. Our optical flow is a dense mapping, $\mathbf{w} = (u(\mathbf{x}), v(\mathbf{x}), 1)^T$, between the pixels of every two consecutive images, where $u(\mathbf{x})$ and $v(\mathbf{x})$ are the x and y displacements in the 3D volume, respectively. We assume that each frame is at distance 1 in time from the previous and following frames.

The Spatial gradient of the image is given by $\nabla I = (I_x, I_y)^T$, with I_x, I_y the first order derivatives in x and y. The Spatial gradient of the optical flow, between two frames, is defined as $\nabla u = (u_x, u_y)^T$ for the *Spatial method* while the spatio-Temporal gradient in the *Temporal method* is $\nabla u = (u_x, u_y, u_t)^T$. Since the abstract framework is identical in both cases, we keep the same notation for both methods. This distinction appears later in the numerical scheme.

We suppose that the problem is not continuous, therefore, the brightness constancy assumption is expressed in its nonlinear form as $I(\mathbf{x} + \mathbf{w}) - I(\mathbf{x}) = 0$. Then, according

to this notation,the energy functional reads as

$$E(\mathbf{w}) = \int_{\Omega} \Psi\left(\left(I(\mathbf{x}+\mathbf{w}) - I(\mathbf{x})\right)^2\right) \mathbf{dx} + \gamma \int_{\Omega} \Psi\left(\left\|\nabla I(\mathbf{x}+\mathbf{w}) - \nabla I(\mathbf{x})\right\|^2\right) \mathbf{dx} + \alpha \int_{\Omega} \Psi\left(\|\nabla u\|^2 + \|\nabla v\|^2\right) \mathbf{dx}, \tag{4.1}$$

with $\Psi\left(s^2\right) = \sqrt{s^2 + \epsilon^2}$ is a robust function and $\epsilon := 0.001$ is a prefixed small constant to ensure that Ψ is strictly convex.

The energy model of Brox *et al.* uses brightness and gradient constancy assumptions in the data term and a TV scheme for smoothing. It depends on the γ and α parameters for controlling the gradient and the smoothness strength, respectively.

Note that this energy functional (eq. 4.1) is different respect to the model presented in [Brox04]. There, the brightness and gradient constancy terms were included inside the same Ψ function. However, we consider better justified that the model separates these two assumptions, as proposed in Bruhn and Weickert [Bruhn05a].

4.1.1 Minimizing the Energy Functional

The solution of the previous energy model can be found by solving the associated Euler-Lagrange equations, given by

$$\begin{aligned}
0 =& \Psi'_D \cdot (I(\mathbf{x}+\mathbf{w}) - I(\mathbf{x})) \cdot I_x(\mathbf{x}+\mathbf{w}) \\
&+ \gamma\, \Psi'_G \cdot ((I_x(\mathbf{x}+\mathbf{w}) - I_x(\mathbf{x})) \cdot I_{xx}(\mathbf{x}+\mathbf{w}) + (I_y(\mathbf{x}+\mathbf{w}) - I_y(\mathbf{x})) \cdot I_{xy}(\mathbf{x}+\mathbf{w})) \\
&- \alpha \operatorname{div}\left(\Psi'_S \cdot \nabla u\right), \\
0 =& \Psi'_D \cdot (I(\mathbf{x}+\mathbf{w}) - I(\mathbf{x})) \cdot I_y(\mathbf{x}+\mathbf{w}) \\
&+ \gamma\, \Psi'_G \cdot ((I_x(\mathbf{x}+\mathbf{w}) - I_x(\mathbf{x})) \cdot I_{xy}(\mathbf{x}+\mathbf{w}) + (I_y(\mathbf{x}+\mathbf{w}) - I_y(\mathbf{x})) \cdot I_{yy}(\mathbf{x}+\mathbf{w})) \\
&- \alpha \operatorname{div}(\Psi'_S \cdot \nabla v),
\end{aligned} \tag{4.2}$$

with $\Psi'(s^2) = \frac{1}{2\sqrt{s^2+\epsilon^2}}$. In order to simplify the equations, we use the following notation:

$$\begin{aligned}
\Psi'_D :=& \Psi'\left(\left(I(\mathbf{x}+\mathbf{w}) - I(\mathbf{x})\right)^2\right), \\
\Psi'_G :=& \Psi'\left(\left\|\nabla I(\mathbf{x}+\mathbf{w}) - \nabla I(\mathbf{x})\right\|^2\right), \\
\Psi'_S :=& \Psi'\left(\|\nabla u\|^2 + \|\nabla v\|^2\right).
\end{aligned} \tag{4.3}$$

The above equations are nonlinear due to the argument $\mathbf{w}$ and function Ψ'. We need to linearize the equations for avoiding this problem; so, we enclose the numerical scheme in two fixed point iterations. We introduce the first index (outer iterations), m, to remove

the nonlinearity in $\mathbf{w}$, using first order Taylor expansions; and the inner iterations, n, to account with the nonlinearities for the Ψ' functions.

$$\begin{aligned} I(\mathbf{x}+\mathbf{w}^{m+1}) \approx & I(\mathbf{x}+\mathbf{w}^{m}) + I_x(\mathbf{x}+\mathbf{w}^{m})du^m + I_y(\mathbf{x}+\mathbf{w}^{m})dv^m \\ I_x(\mathbf{x}+\mathbf{w}^{m+1}) \approx & I_x(\mathbf{x}+\mathbf{w}^{m}) + I_{xx}(\mathbf{x}+\mathbf{w}^{m})du^m + I_{xy}(\mathbf{x}+\mathbf{w}^{m})dv^m \\ I_y(\mathbf{x}+\mathbf{w}^{m+1}) \approx & I_y(\mathbf{x}+\mathbf{w}^{m}) + I_{xy}(\mathbf{x}+\mathbf{w}^{m})du^m + I_{yy}(\mathbf{x}+\mathbf{w}^{m})dv^m, \end{aligned} \tag{4.4}$$

with $\mathbf{w}^m = (u^m, v^m)^T$, $du^m = u^{m+1} - u^m$ and $dv^m = v^{m+1} - v^m$. We assume that $\mathbf{w}^m$ is a close approximation to our unknowns $\mathbf{w}^{m+1}$.

We use *motion increments* (du^m, dv^m) to calculate the optical flows [Mémin98]. These are incrementally updated from the motion increment as $u^{m+1} = u^m + du^m$, $v^{m+1} = v^m + dv^m$.

$$\begin{aligned} 0 =& (\Psi'_D)^{m,n} \cdot \big(I(\mathbf{y}) + I_x(\mathbf{y})du^{m,n+1} + I_y(\mathbf{y})dv^{m,n+1} - I(\mathbf{x})\big) \cdot I_x(\mathbf{y}) \\ & + \gamma\,(\Psi'_G)^{m,n} \cdot \Big(\big(I_x(\mathbf{y}) + I_{xx}(\mathbf{y})du^{m,n+1} + I_{xy}(\mathbf{y})dv^{m,n+1} - I_x(\mathbf{x})\big) \cdot I_{xx}(\mathbf{y}) \\ & \quad + \big(I_y(\mathbf{y}) + I_{xy}(\mathbf{y})du^{m,n+1} + I_{yy}(\mathbf{y})dv^{m,n+1} - I_y(\mathbf{x})\big) \cdot I_{xy}(\mathbf{y}^{m,n})\Big) \\ & - \alpha\,\mathrm{div}\left((\Psi'_S)^{m,n} \cdot \nabla(u^{m,n} + du^{m,n+1})\right) \\ 0 =& (\Psi'_D)^{m,n} \cdot \big(I(\mathbf{y}) + I_x(\mathbf{y})du^{m,n+1} + I_y(\mathbf{y})dv^{m,n+1} - I(\mathbf{x})\big) \cdot I_x(\mathbf{y}) \\ & + \gamma\,(\Psi'_G)^{m,n} \cdot \Big(\big(I_x(\mathbf{y}) + I_{xx}(\mathbf{y})du^{m,n+1} + I_{xy}(\mathbf{y})dv^{m,n+1} - I_x(\mathbf{x})\big) \cdot I_{xx}(\mathbf{y}) \\ & \quad + \big(I_y(\mathbf{y}) + I_{xy}(\mathbf{y})du^{m,n+1} + I_{yy}(\mathbf{y})dv^{m,n+1} - I_y(\mathbf{x})\big) \cdot I_{xy}(\mathbf{y}^{m,n})\Big) \\ & - \alpha\,\mathrm{div}\left((\Psi'_S)^{m,n} \cdot \nabla(v^{m,n} + dv^{m,n+1})\right), \end{aligned} \tag{4.5}$$

with $\mathbf{y} = \mathbf{x} + \mathbf{w}^{m,n}$.

The system of equations (4.5) can be efficiently solved using the SOR method. The unknowns $du^{m,n+1}$ and $dv^{m,n+1}$, in pixel (i, j, t), are expressed in terms of the remaining terms, and their values are iteratively updated until the method converges to a steady state solution. In this sense, we introduce an additional fixed point iteration scheme, s, for the SOR method.

Partial derivatives are approximated using central differences. The discretization of the divergence is separated in three variables: $\mathrm{div}\left((\Psi'_S)^{m,n} \cdot \nabla(u^{m,n} + du^{m,n+1})\right) = \mathrm{div}\left((\Psi'_S)^{m,n} \cdot \nabla u^{m,n}\right) + \mathrm{div}\left((\Psi'_S)^{m,n} \cdot \nabla du^{m,n+1}\right) \approx div\_u + (div\_du - div\_d \cdot du^{m,n+1}_{i,j,t})$, where div_u discretizes the first divergence term, div_d and div_du correspond to the second term. In the second term, div_du stands for the values corresponding to the neighbors of du, and div_d stands for the coefficients accompanying du at the current

pixel, $du_{i,j,t}^{m,n+1}$. These variables are given by the following expressions:

$$\begin{aligned} div\_u :=& \frac{(\Psi'_S)_{i+1,j,t} + (\Psi'_S)_{i,j,t}^{m,n}}{2} \left(u_{i+1,j,t}^{m,n} - u_{i,j,t}^{m,n}\right) + \frac{(\Psi'_S)_{i-1,j,t} + (\Psi'_S)_{i,j,t}^{m,n}}{2} \left(u_{i-1,j,t}^{m,n} - u_{i,j,t}^{m,n}\right) + \\ & \frac{(\Psi'_S)_{i,j+1,t} + (\Psi'_S)_{i,j,t}^{m,n}}{2} \left(u_{i,j+1,t}^{m,n} - u_{i,j,t}^{m,n}\right) + \frac{(\Psi'_S)_{i,j-1,t} + (\Psi'_S)_{i,j,t}^{m,n}}{2} \left(u_{i,j-1,t}^{m,n} - u_{i,j,t}^{m,n}\right) + \\ & \frac{(\Psi'_S)_{i,j,t+1} + (\Psi'_S)_{i,j,t}^{m,n}}{2} \left(u_{i,j,t+1}^{m,n} - u_{i,j,t}^{m,n}\right) + \frac{(\Psi'_S)_{i,j,t-1} + (\Psi'_S)_{i,j,t}^{m,n}}{2} \left(u_{i,j,t-1}^{m,n} - u_{i,j,t}^{m,n}\right), \end{aligned} \tag{4.6}$$

$$\begin{aligned} div\_du :=& \frac{(\Psi'_S)_{i+1,j,t} + (\Psi'_S)_{i,j,t}^{m,n}}{2} du_{i+1,j,t}^{m,n+1} + \frac{(\Psi'_S)_{i-1,j,t} + (\Psi'_S)_{i,j,t}^{m,n}}{2} du_{i-1,j,t}^{m,n+1} + \\ & \frac{(\Psi'_S)_{i,j+1,t} + (\Psi'_S)_{i,j,t}^{m,n}}{2} du_{i,j+1,t}^{m,n+1} + \frac{(\Psi'_S)_{i,j-1,t} + (\Psi'_S)_{i,j,t}^{m,n}}{2} du_{i,j-1,t}^{m,n+1} + \\ & \frac{(\Psi'_S)_{i,j,t+1} + (\Psi'_S)_{i,j,t}^{m,n}}{2} du_{i,j,t+1}^{m,n+1} + \frac{(\Psi'_S)_{i,j,t-1} + (\Psi'_S)_{i,j,t}^{m,n}}{2} du_{i,j,t-1}^{m,n+1}, \end{aligned} \tag{4.7}$$

$$\begin{aligned} div\_d :=& \frac{(\Psi'_S)_{i+1,j,t} + (\Psi'_S)_{i,j,t}^{m,n}}{2} + \frac{(\Psi'_S)_{i-1,j,t} + (\Psi'_S)_{i,j,t}^{m,n}}{2} + \\ & \frac{(\Psi'_S)_{i,j+1,t} + (\Psi'_S)_{i,j,t}^{m,n}}{2} + \frac{(\Psi'_S)_{i,j-1,t} + (\Psi'_S)_{i,j,t}^{m,n}}{2} + \\ & \frac{(\Psi'_S)_{i,j,t+1} + (\Psi'_S)_{i,j,t}^{m,n}}{2} + \frac{(\Psi'_S)_{i,j,t-1} + (\Psi'_S)_{i,j,t}^{m,n}}{2}. \end{aligned} \tag{4.8}$$

These expressions are the same for the other component of the optical flow, changing the roles of u and v.

The last two terms of equations (4.6), (4.7) and (4.8), in italics, correspond to the Temporal regularization of the optical flow. This is implemented in the Temporal method and removed in the Spatial one. The finite difference scheme in the Temporal method is computed using information from the previous, $m-1$, and following, $m+1$, frames. If we define $\mathbf{y} = \mathbf{x} + \mathbf{w}^{m,n}$ and separate the parts of the equation that remain constant during the SOR iterations, we may define the following variables:

$$\begin{aligned} Au :=& -\left(\Psi'_D\right)^{m,n} \left(I(\mathbf{y}) - I(\mathbf{x})\right) I_x(\mathbf{y}) + \alpha\, div\_u, \\ & -\gamma \left(\Psi'_G\right)^{m,n} \left(\left(I_x(\mathbf{y}) - I_x(\mathbf{x})\right) I_{xx}(\mathbf{y}) + \left(I_y(\mathbf{y}) - I_y(\mathbf{x})\right) I_{xy}(\mathbf{y})\right), \\ Av :=& -\left(\Psi'_D\right)^{m,n} \left(I(\mathbf{y}) - I(\mathbf{x})\right) I_y(\mathbf{y}) + \alpha\, div\_v \\ & -\gamma \left(\Psi'_G\right)^{m,n} \left(\left(I_x(\mathbf{y}) - I_x(\mathbf{x})\right) I_{xy}(\mathbf{y}) + \left(I_y(\mathbf{y}) - I_y(\mathbf{x})\right) I_{yy}(\mathbf{y})\right), \\ Du :=& \left(\Psi'_D\right)^{m,n} I_x^2(\mathbf{y}) + \gamma \left(\Psi'_G\right)^{m,n} \left(I_{xx}^2(\mathbf{y}) + I_{xy}^2(\mathbf{y})\right) + \alpha\, div\_d, \\ Dv :=& \left(\Psi'_D\right)^{m,n} I_y^2(\mathbf{y}) + \gamma \left(\Psi'_G\right)^{m,n} \left(I_{yy}^2(\mathbf{y}) + I_{xy}^2(\mathbf{y})\right) + \alpha\, div\_d, \\ D :=& \left(\Psi'_D\right)^{m,n} I_x(\mathbf{y}) I_y(\mathbf{y}) + \gamma \left(\Psi'_G\right)^{m,n} \left(I_{xx}(\mathbf{y}) + I_{yy}(\mathbf{y})\right) I_{xy}(\mathbf{y}). \end{aligned} \tag{4.9}$$

In order to compute expressions like $I(\mathbf{x}+\mathbf{w}^{m,n})$, we use bicubic interpolation. Putting all together, we arrive to the SOR scheme, which is given by

$$\begin{aligned} du^{m,n,s+1} :=& \frac{(1-w)\,du^{m,n,s} + w\,(Au - D\cdot dv^{m,n,s+1} + \alpha\, div\_du)}{Du}, \\ dv^{m,n,s+1} :=& \frac{(1-w)\,dv^{m,n,s} + w\,(Av - D\cdot du^{m,n,s+1} + \alpha\, div\_dv)}{Dv}, \end{aligned} \tag{4.10}$$

with $w \in (0,2)$ the SOR relaxation parameter. In our implementation, we choose $w = 1.9$ by default.

This numerical approximation is calculated until the method converges to a steady state solution or it exceeds a maximum number of iterations. The stopping criterion is

$$\frac{1}{N}\sum_{i,j,t}\left(du_{i.j,t}^{s+1} - du_{i.j,t}^{s}\right)^2 + \left(dv_{i.j,t}^{s+1} - dv_{i.j,t}^{s}\right)^2 < \varepsilon^2, \tag{4.11}$$

with N the number of pixels in all frames and ε the stopping criterion threshold. This is different from the original article [Brox04], where a fixed number of SOR iterations is used. Once it has converged, we go to the next inner iteration, $n+1$, and restart the variables in (4.9).

4.1.2 Pyramidal Structure

In order to estimate large displacements, we embed the optical flow method in a pyramidal structure. We follow the same strategy presented in a previous IPOL article [ML13] and reproduce here the basic ideas.

The algorithm creates a pyramid of down-sampled images. The pyramid is created by reducing the images by a factor $\eta \in (0,1)$. Before downsampling, the images are smoothed with a Gaussian kernel of a standard deviation that depends on η. For a set of scales $s = 0, 1, \ldots, N_{scales} - 1$, the pyramid of images is built as

$$I^s(\eta\mathbf{x}) := G_\sigma * I^{s-1}(\mathbf{x}). \tag{4.12}$$

After the convolution, the images are sampled using bicubic interpolation. The value of σ depends on η and is calculated as

$$\sigma(\eta) := \sigma_0\sqrt{\eta^{-2}-1},\ with\ \sigma_0 := 0.6. \tag{4.13}$$

Then, starting at the coarsest scale, the system of equations is solved in each scale to get successive approximations of the optical flow. Every intermediate solution is used as

initialization in the following scale. To transfer the values from a coarser scale, the flow field is updated as

$$\begin{aligned} u^{s-1}(\mathbf{x}) &:= \frac{1}{\eta} u^s(\eta \mathbf{x}) \\ v^{s-1}(\mathbf{x}) &:= \frac{1}{\eta} v^s(\eta \mathbf{x}). \end{aligned} \tag{4.14}$$

4.1.3 Multi-Channel Robust Optical Flow

Next, we show the main modifications required in the Spatial version of the previous energy model to adapt it to a multi-channel scheme.

Let $I_1^c, I_2^c : \Omega \subset \mathbb{R}^2 \to \mathbb{R}^c$ be an image sequence, with $\mathbf{x} = (x, y)^T \in \Omega$, $\{I^c\}_{c=1,\dots,C}$ and C the number of channels. Once again, our optical flow is $\mathbf{w}(\mathbf{x}) = (u(\mathbf{x}), v(\mathbf{x}), 1)^T$ and the vector fields are $u(\mathbf{x})$ and $v(\mathbf{x})$. However, we now denote the Spatial gradient of the image with $\nabla I = \{(I_x^c, I_y^c)^T\}_{c=1,\dots,C}$, where I_x^c, I_y^c are the first order derivatives in x and y for each information channel. Then, we express our multi-channel model as

$$\begin{aligned} E(\mathbf{w}) = &\int_\Omega \Psi \left(\sum_{c=1}^{C} \left(I_2^c(\mathbf{x}+\mathbf{w}) - I_1^c(\mathbf{x}) \right)^2 \right) + \gamma \int_\Omega \Psi \left(\sum_{c=1}^{C} \left\| \nabla I_2^c(\mathbf{x}+\mathbf{w}) - \nabla I_1^c(\mathbf{x}) \right\|^2 \right) \\ &+ \alpha \int_\Omega \Psi \left(\left(\|\nabla u\|^2 + \|\nabla v\|^2 \right) \right) \mathbf{dx}, \end{aligned} \tag{4.15}$$

with $\Psi(s^2) = \sqrt{s^2 + \epsilon^2}$ and $\epsilon := 0.001$ like in the previous functional (4.1). Again, we assume that the brightness constancy assumption is also fulfilled in the multi-channel scheme.

The minimum of the functional is found by solving the following Euler-Lagrange equations:

$$\begin{aligned} 0 =& \Psi'_D \cdot \left(\sum_{c=1}^{C} \left(I_2^c(\mathbf{x}+\mathbf{w}) - I_1^c(\mathbf{x}) \right) \cdot I_{2x}^c(\mathbf{x}+\mathbf{w}) \right) \\ &+ \gamma\, \Psi'_G \cdot \left(\sum_{c=1}^{C} \left(I_{2x}^c(\mathbf{x}+\mathbf{w}) - I_{1x}^c(\mathbf{x}) \right) \cdot I_{2xx}^c(\mathbf{x}+\mathbf{w}) + \left(I_{2y}^c(\mathbf{x}+\mathbf{w}) - I_{1y}^c(\mathbf{x}) \right) \cdot I_{2xy}^c(\mathbf{x}+\mathbf{w}) \right) \\ &- \alpha\, \mathrm{div} \left(\Psi'_S \cdot \nabla u \right), \\ 0 =& \Psi'_D \cdot \left(\sum_{c=1}^{C} \left(I_2^c(\mathbf{x}+\mathbf{w}) - I_1^c(\mathbf{x}) \right) \cdot I_{2y}^c(\mathbf{x}+\mathbf{w}) \right) \\ &+ \gamma\, \Psi'_G \cdot \left(\sum_{c=1}^{C} \left(I_{2x}^c(\mathbf{x}+\mathbf{w}) - I_{1x}^c(\mathbf{x}) \right) \cdot I_{2xy}^c(\mathbf{x}+\mathbf{w}) + \left(I_{2y}^c(\mathbf{x}+\mathbf{w}) - I_{1y}^c(\mathbf{x}) \right) \cdot I_{2yy}^c(\mathbf{x}+\mathbf{w}) \right) \\ &- \alpha\, \mathrm{div}(\Psi'_S \cdot \nabla v), \end{aligned} \tag{4.16}$$

with $\Psi'(s^2) = \frac{1}{2\sqrt{s^2+\epsilon^2}}$. Notice that the difference with respect to the original model resides in the summation of image channels included in both constancy assumptions of the data term. As a consequence, the influence of this term is increased proportional to the number of channels. We compensate this situation by adapting the smoothness parameter α, therefore, $\alpha = \alpha' \cdot C$, being α' an input parameter. An example of this multi-channel scheme can be seen in Fig. 4.1. We use the following abbreviations:

$$\begin{aligned}
\Psi'_D :=& \Psi' \left(\sum_{c=1}^{C} \left(I_2^c(\mathbf{x}+\mathbf{w}) - I_1^c(\mathbf{x})\right)^2 \right), \\
\Psi'_G :=& \Psi' \left(\sum_{c=1}^{C} \left\|\nabla I_2^c(\mathbf{x}+\mathbf{w}) - \nabla I_1^c(\mathbf{x})\right\|^2 \right), \\
\Psi'_S :=& \Psi' \left(\|\nabla u\|^2 + \|\nabla v\|^2 \right).
\end{aligned} \tag{4.17}$$

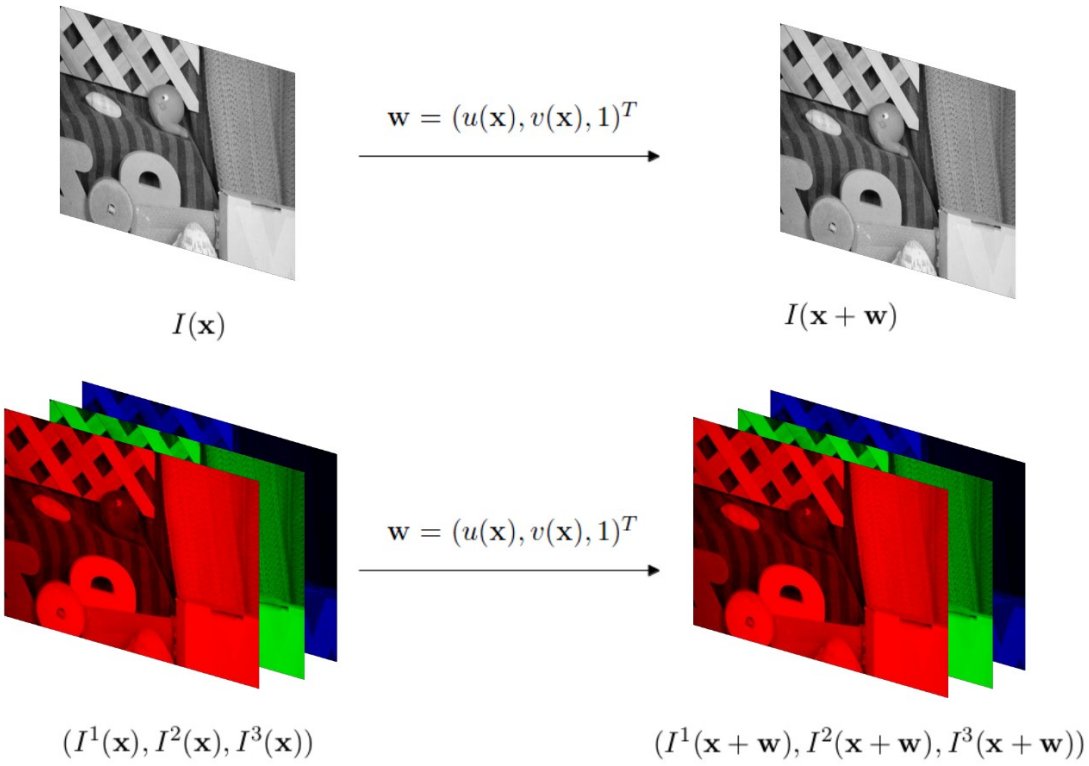


Figure 4.1: Notation for grayscale and color images using frames 10 and 11 from the *RubberWhale* sequence.

As in the grayscale model, we use centered finite differences to discretize the system

and solve the system of equations (4.16) with the SOR method. The non-linearity of the above formulas (4.17) are linearized by using again two fixed point iterations. We also embedded it in a multiscale strategy to allow detecting large displacements. The warping of $I_2^c(\mathbf{x}+\mathbf{w})$ is approximated using Taylor series and bicubic interpolation.

4.2 Experimental Results

In this section, we examine the behavior of the method for some standard sequences using their grayscale images. In particular, we use Yosemite (with and without clouds) and two sequences from the *Middlebury* benchmark database (*RubberWhale* and *Urban2*). In our experiments, we have looked for the best α and γ parameters (table 4.2) that provide the best results presented at figure 4.2. The values for the remaining parameters are: $\eta = 0.75$, $\varepsilon = 0.0001$, $inner\_iterations = 1$, $outer\_iterations = 38$, and N_{scales} is automatically calculated so that the coarsest scale works with images around 16×16 pixels. The optical flow representation is given by the IPOL color scheme.

We observe that, in general, the L^1 functionals and the TV regularization scheme create piecewise continuous flow fields and provide a good preservation of the motion discontinuities. However, the method also creates rounded shapes at these boundaries. Besides, we also observe an accurate behavior dealing with the translational motion (2 pixels to the right) of the sky in the *Yosemite with Clouds* sequence despite of the illumination changes in the clouds. In this case, the benefits of the gradient constancy term are clear.

Table 4.1 shows the Average Angular Error (AAE) and Average End-point Error (EPE) for the Spatial and Temporal methods, respectively. The AAE and EPE are calculated as in the *Middlebury* benchmarks [Baker07b]. The present results are quite similar to the ones of the original paper [Brox04] for the Yosemite sequences. According to these results, the Spatial approach is worst regarding to the Temporal method. The flow in these sequences is very smooth, so it is clearly benefited from the spatio-temporal continuous regularization term.

Table 4.1: Numerical results for the Spatial and Temporal solutions of figure 4.2. We remark in boldface the best error for each sequence.

Sequence	Spatial AAE	Spatial EPE	Temporal AAE	Temporal EPE
Yosemite	1.587^o	0.075	$\mathbf{1.297^o}$	**0.061**
Yosemite with Clouds	2.367^o	0.101	$\mathbf{1.927^o}$	**0.079**
RubberWhale	$\mathbf{3.467^o}$	**0.103**	4.798^o	0.152
Urban2	$\mathbf{2.803^o}$	**0.395**	5.823^o	0.560

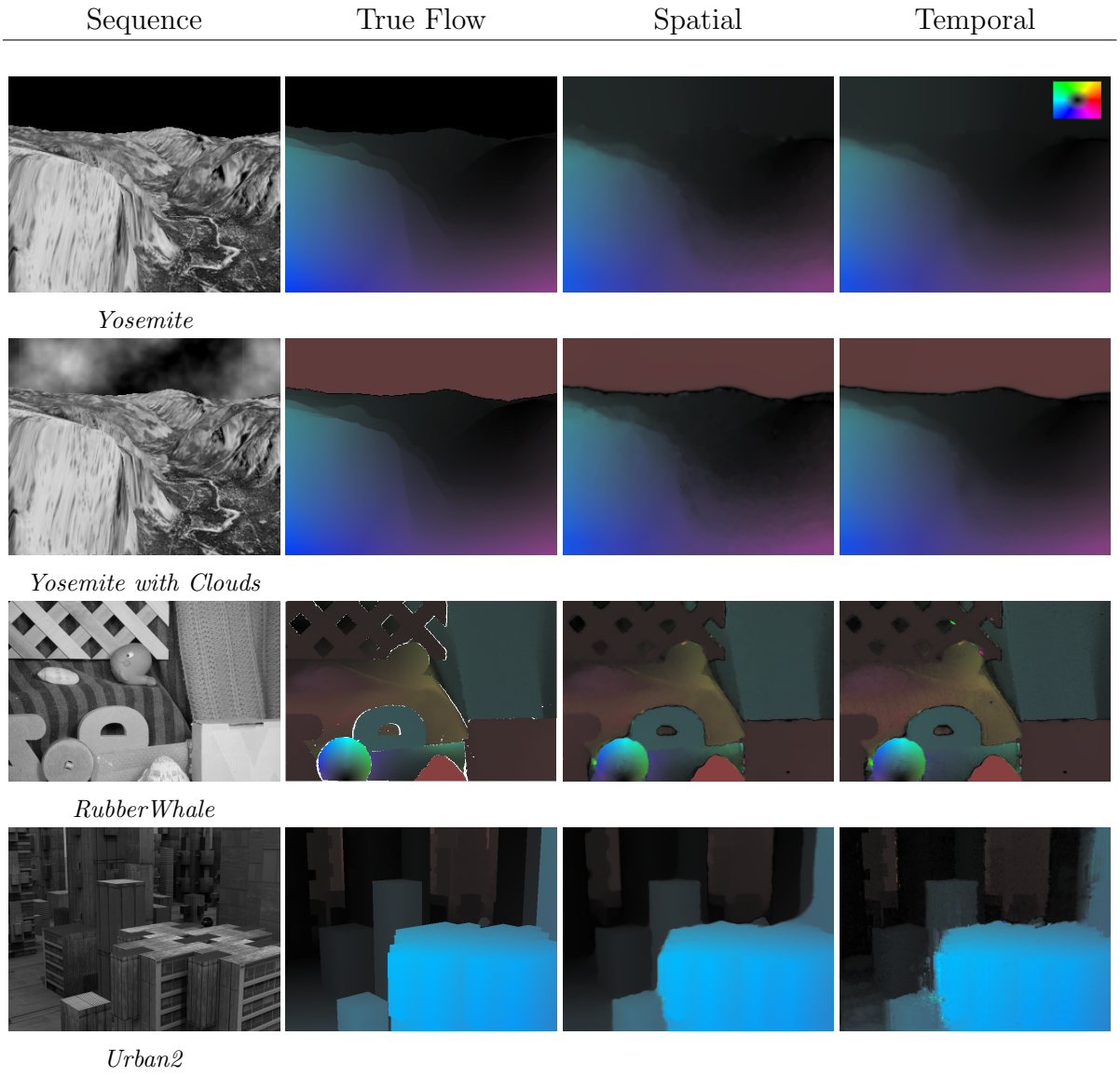
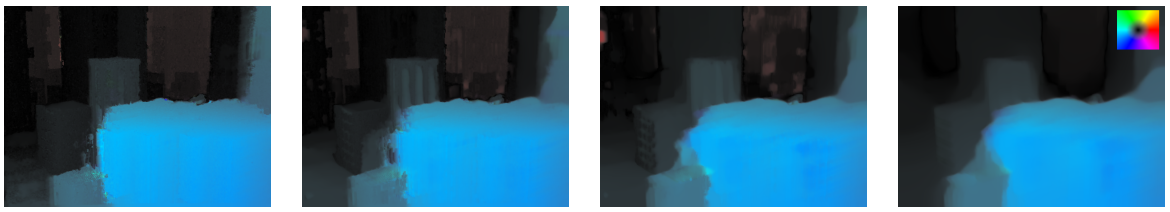


Figure 4.2: Results for the *Yosemite*, *Yosemite with Clouds*, *RubberWhale* and *Urban2* sequences. First column shows frame 6 for *Yosemite* and *Yosemite with Clouds*, and frame 10 for *RubberWhale* and *Urban2*. Second column shows the corresponding true flows. Third and fourth columns show the results for the Spatial and Temporal methods, respectively.



$\alpha = 2$ $\alpha = 7$ $\alpha = 15$ $\alpha = 30$

Figure 4.3: Results for the *Urban2* sequence using the Temporal method. We depict the influence of the α parameter and how it increases the rounded shapes in the discontinuities.

Table 4.2: Best α and γ for the best Spatial and Temporal solutions of figure 4.2.

	Spatial		Temporal	
	α	γ	α	γ
Yosemite	50	2	27	2
Yosemite with Clouds	145	15	93	15
RubberWhale	185	60	91	60
Urban2	30	2	2	2

Nevertheless, in the cases of *RubberWhale* and *Urban2* the situation is the opposite. The solutions of the Temporal method are worse than its Spatial counterpart. The best flow achieved for *RubberWhale* presents piecewise continuous motion fields but, it also mistakes some motion directions and has problems at the flow discontinuities. Moreover, the Temporal regularization worsen the results in *Urban2*, where the maximum motion is about 22 pixels. The discontinuities of the optical flows are strongly deteriorated for bigger values of α as we observe in figure 4.3. The effects of continuous Temporal smoothing scheme are negative and, and it is necessary to use small values of α to obtain results similar to the Spatial method. In these sequences the presence of flow discontinuities and large displacements is more important, therefore the continuous Temporal scheme is not suitable in these cases. As a consequence, it is better to use a nonlinear Temporal smoothing scheme, like in [Salgado06] or the method explained in chapter 3.

On the other hand, figure 4.4 compares the AAE and EPE evolutions with respect to α for both approaches. As we expected, the results of the Temporal method for the *Yosemite* sequence are better than the Spatial method. The regularization is more important with the Temporal scheme, thus smaller α values attain smaller AAE and EPE. We also observe that the Temporal results degrade faster for increasing values of α. It crosses the Spatial curve and then diverges. This may occur because the Temporal regularization tends to spoil faster the flow discontinuities.

Interestingly, we observe that the best AAE and EPE are obtained at different values of α for the same sequence. The best AAE is obtained with a larger α. This is justified because a larger α creates smoother flows, which may be better aligned with the directions

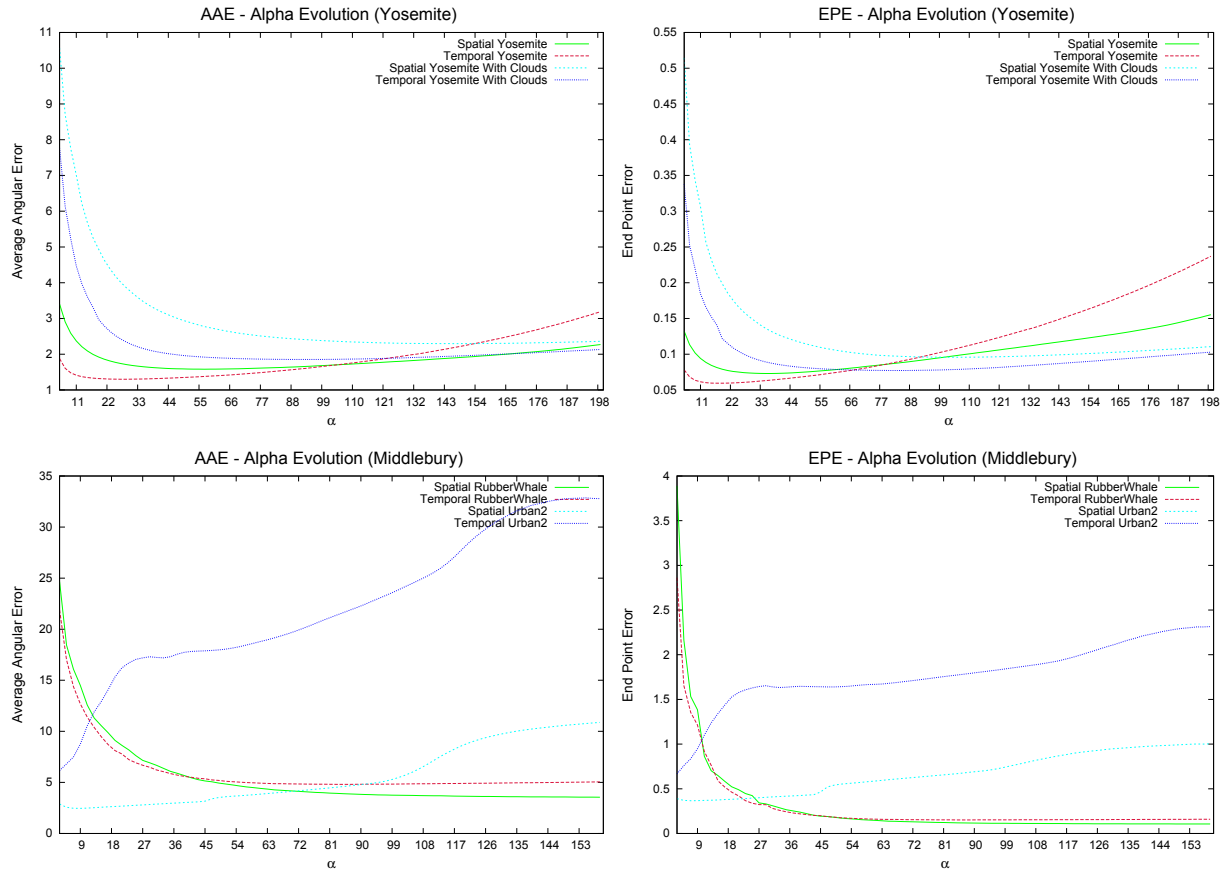


Figure 4.4: First row, the AAE and EPE for *Yosemite* and *Yosemite with Clouds* sequences; second row, the AAE and EPE for *RubberWhale* and *Urban2*.

of the true flow, but with smaller magnitudes. In the bottom row of figure 4.4, we observe that the Temporal method always provides worse results in the *RubberWhale* and *Urban2* sequences. Note that the Temporal curve for *Urban2* depicts a strong deterioration for increasing values of α.

In the graphics of figure 4.5, we compare the Spatial and Temporal methods, frame by frame, for the *Yosemite with Clouds* sequence. Most of the optical flow errors in the Temporal method remain below the Spatial errors. When α is big, $\alpha = 350$, the errors in the Temporal method exceed the Spatial ones.

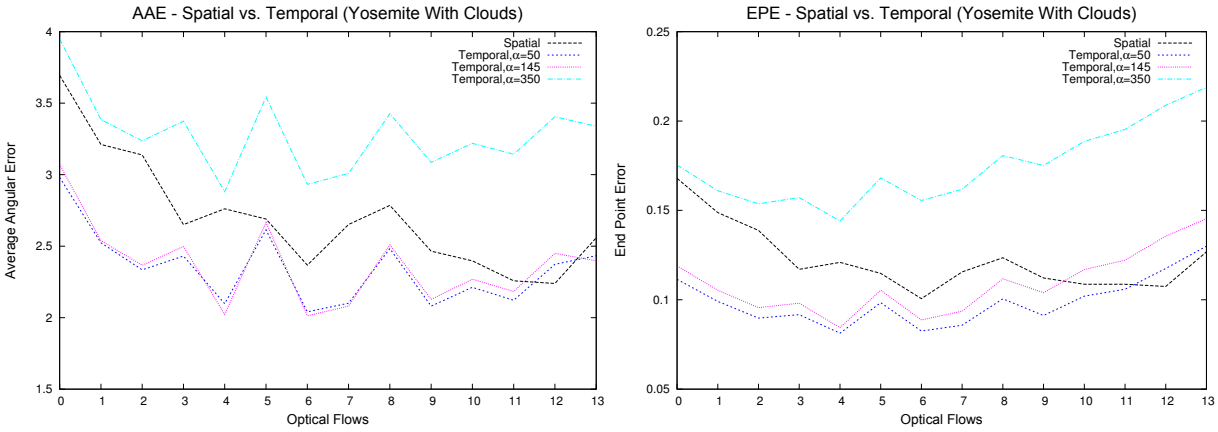


Figure 4.5: Comparison of the Spatial and Temporal methods: AAE (left) and EPE (right) for the *Yosemite with Clouds* sequence in every frame.

Table 4.3 depicts the error evolution with respect to the η parameter. In this experiment, we have used the best α and γ parameters of table 4.2. We note that the AAE and EPE do not significantly improve for biggest values of η and N_{scales}.

In the Yosemite sequences, we observe a difference between our results and the solutions of the original work [Brox04], where the best flows were obtained for $\eta = 0.95$, with a large number of scales and a lot of inner and outer iterations. In our case, the errors are hardly improved with η. Even for values of $\eta = 0.1$ the results are satisfactory. This is interesting because this means that the method can find a very good solution using only two scales, and let the coarsest scale start with a very small image size. Moreover, the *RubberWhale* and *Urban2* sequences present a similar behavior, but the results are more noticeable for $\eta = 0.1$ and $\eta = 0.25$.

Next, the figures 4.6, 4.7 and 4.8 show the AAE evolution with respect to α and γ for *Yosemite with Clouds*, *RubberWhale* and *Urban2* sequences, respectively. Note that the graphics are very stable for a large range of values, especially for *Yosemite with Clouds* and *RubberWhale*. We do not include Yosemite's graphic because it is very similar to *Yosemite with Clouds*.

Table 4.3: AAE and EPE results for different values of η and N_{scales}.

			Spatial		**Temporal**	
Sequence	η	N_{scales}	AAE	EPE	AAE	EPE
Yosemite	0.1	2	1.602^o	0.076	1.314^o	0.062
	0.25	2	1.605^o	0.076	1.310^o	0.062
	0.5	4	1.594^o	0.075	1.303^o	0.061
	0.65	7	1.589^o	0.075	1.300^o	0.061
	0.75	10	1.586^o	0.075	1.297^o	0.061
	0.85	17	1.586^o	0.075	1.297^o	0.061
	0.95	54	1.586^o	0.075	1.297^o	0.061
Yosemite with Clouds	0.1	2	2.461^o	0.101	2.182^o	0.091
	0.25	2	2.382^o	0.099	2.017^o	0.083
	0.5	4	2.372^o	0.101	1.944^o	0.080
	0.65	5	2.359^o	0.100	1.933^o	0.079
	0.75	10	2.367^o	0.100	1.927^o	0.079
	0.85	17	2.379^o	0.101	1.922^o	0.079
	0.95	54	2.393^o	0.103	1.931^o	0.079
RubberWhale	0.1	2	3.779^o	0.117	5.725^o	0.185
	0.25	3	3.616^o	0.109	5.091^o	0.162
	0.5	5	3.491^o	0.104	4.919^o	0.155
	0.65	8	3.491^o	0.104	4.847^o	0.153
	0.75	12	3.467^o	0.103	4.798^o	0.152
	0.85	20	3.458^o	0.103	4.831^o	0.154
	0.95	63	3.426^o	0.102	4.840^o	0.159
Urban2	0.1	2	3.316^o	0.570	7.592^o	0.910
	0.25	3	2.873^o	0.426	6.110^o	0.698
	0.5	5	2.836^o	0.408	5.787^o	0.614
	0.65	8	2.823^o	0.392	5.931^o	0.603
	0.75	12	2.803^o	0.395	5.823^o	0.560
	0.85	21	2.790^o	0.385	6.043^o	0.626
	0.95	67	2.784^o	0.377	6.170^o	0.623

We use a red curve for showing the best errors for every pair (α, γ). According to the results, the behavior is quite interesting due to the fact that the curves approximate, in the graphics, a straight line in the $\alpha - \gamma$ plane. The relation is $\alpha \simeq 20\gamma$, $\alpha \simeq 10\gamma$, $\alpha \simeq 3\gamma$ and $\alpha \simeq 11\gamma$ in the sequences of *Yosemite*, *Yosemite with Clouds*, *RubberWhale* and *Urban2*, respectively. This means that, if we respect these relations according we increase the parameters, the errors still remain good.

In these graphics, we represent with a green dot the minimum error. We observe an interesting behavior in the *RubberWhale* sequence, where we get slightly better results for even higher values of α and γ, although the improvements are hardly appreciable.

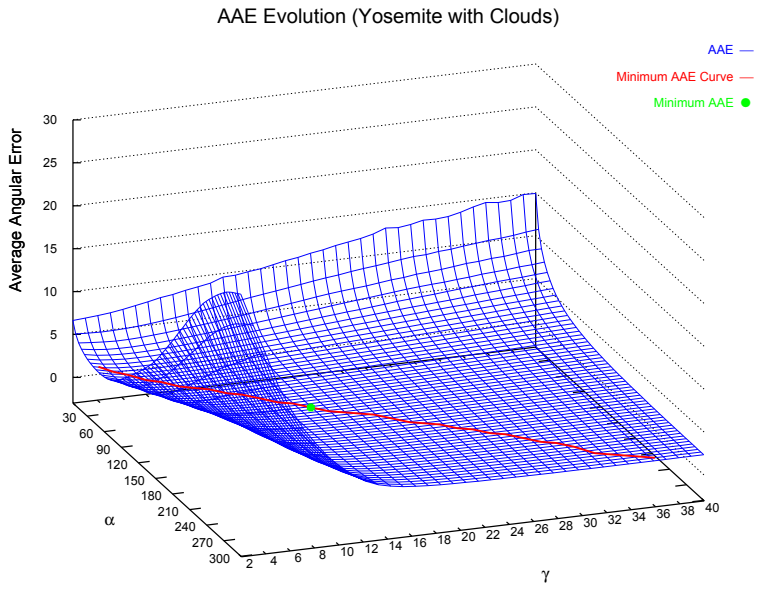


Figure 4.6: *Yosemite with Clouds*. Evolution of AAE with respect to α and γ.

Finally, we show the relation between the *inner_iterations* and *outer_iterations* respect to the AAE in figures 4.9, 4.10 and 4.11 for the same sequences of the previous experiment. We observe a very good stability according these parameters in the graphics. Besides, the method converges very quickly to the final solution. In graphics, we also show several color points for the minimum AAE and the values with an error below 0.5%, 1% and 3% with respect to the minimum. The last three values were calculated for the smallest *inner_iterations* $\times$ *outer_iterations* relation, i.e., the minimum number of total iterations.

After this experiment, we conclude that the *outer_iterations* parameter is more relevant in the convergence of the method. Normally, for *inner_iterations* $= 1$ we obtain very good accuracies; therefore, we may suppose that the inner iterations in these experiments can be integrated in the outer iterations without a loss of precision.

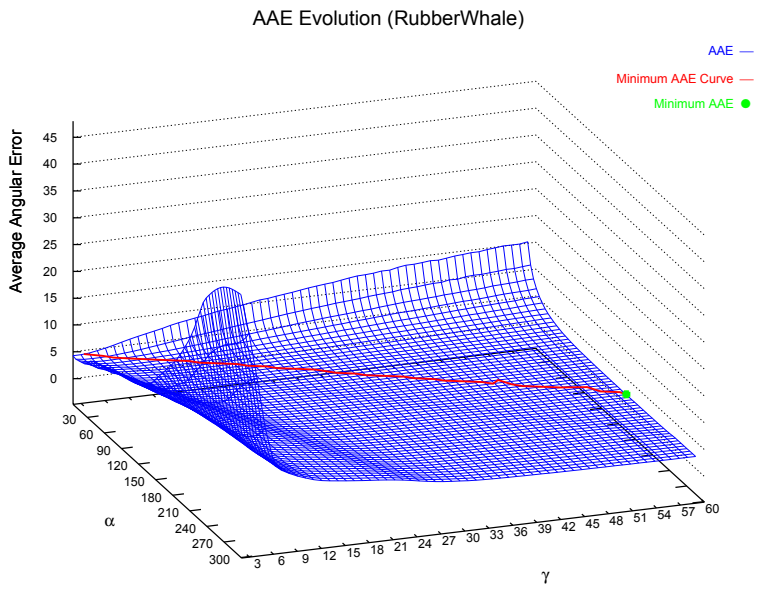


Figure 4.7: *RubberWhale*. Evolution of AAE with respect to α and γ.

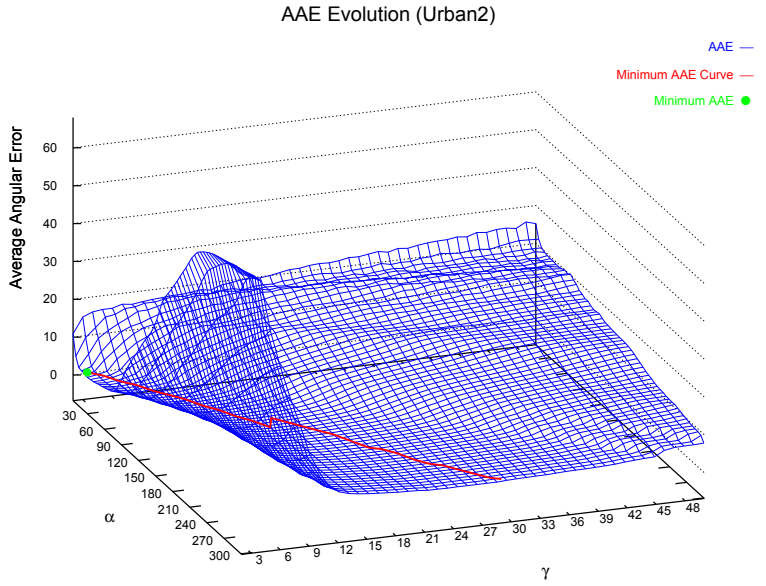


Figure 4.8: *Urban2*. Evolution of AAE with respect to α and γ.

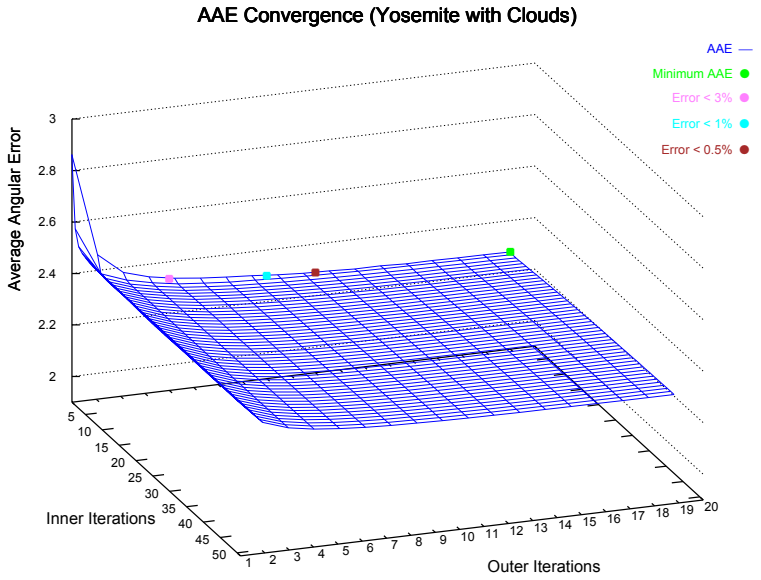


Figure 4.9: *Yosemite with Clouds.* Evolution of AAE with respect to *inner_iterations* and *outer_iterations*.

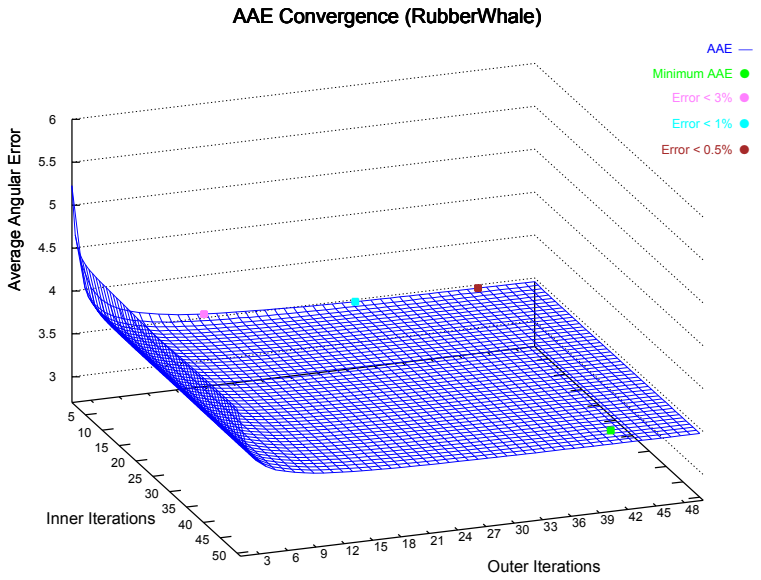


Figure 4.10: *RubberWhale.* Evolution of AAE with respect to *inner_iterations* and *outer_iterations*.

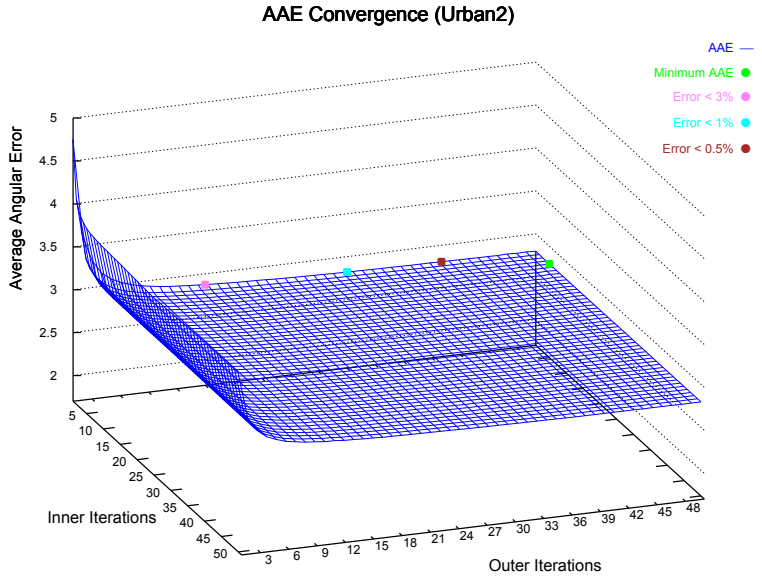


Figure 4.11: *Urban2*. Evolution of AAE with respect to *inner_iterations* and *outer_iterations*.

4.2.1 Examples

Next, we depict the results for the test sequences of the *Middlebury* benchmark database in figure 4.12. As in the previous chapter, we do not use *Dimetrodon* and *Venus* because these sequences do not have enough images for the Temporal method. Table 4.4 show the AAE and EPE obtained for $\alpha = 18$ and $\gamma = 7$ in the Spatial method, and $\alpha = 2.5$, $\gamma = 2$ in the Temporal method. In both cases, we have set the following parameters: $\eta = 0.75$, $\varepsilon = 0.0001$, $inner\_iterations = 1$ and $outer\_iterations = 15$. N_{scales} is automatically calculated so that the coarsest scale works with images around 16 x 16 pixels.

Table 4.4: AAE and EPE for the *Middlebury* test sequences.

Sequences	Spatial AAE	Spatial EPE	Temporal AAE	Temporal EPE
Grove2	2.455^o	0.174	2.569^o	0.184
Grove3	6.481^o	0.693	7.031^o	0.796
Hydrangea	2.442^o	0.200	4.468^o	0.346
RubberWhale	3.696^o	0.111	5.435^o	0.168
Urban2	2.561^o	0.368	5.903^o	0.628
Urban3	4.804^o	0.544	6.681^o	0.784
Dimetrodon	1.663^o	0.086	-	-
Venus	4.599^o	0.292	-	-

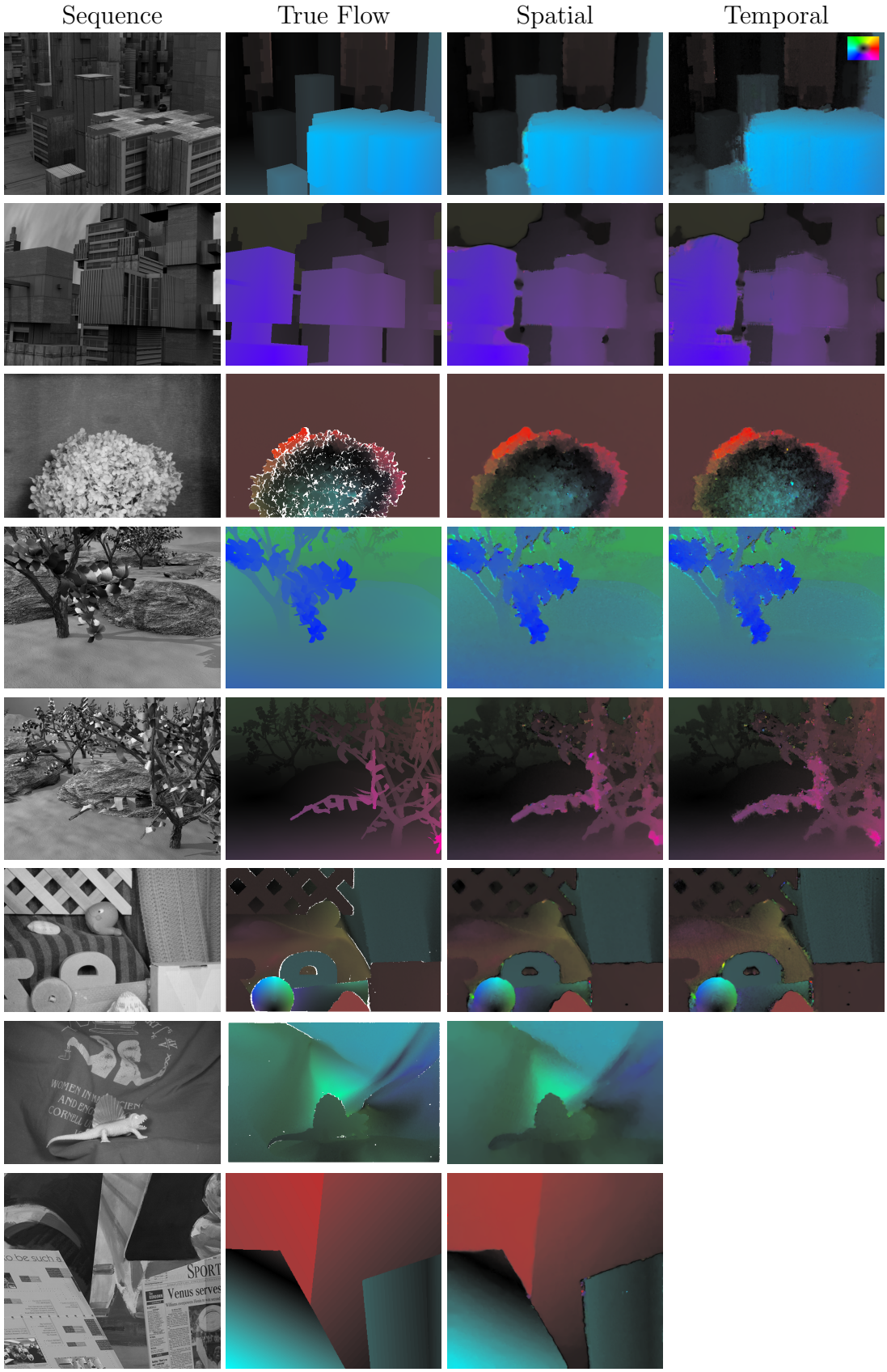


Figure 4.12: Results for the *Middlebury* test sequences using grayscale images.

Finally, in figure 4.13 we show the results for the evaluation sequences using the same parameter configuration (Spatial method).

Figure 4.13: Results for the the *Middlebury* evaluation sequences using grayscale images.

4.2.2 Influence of Color in Optical Flow Estimation

The previous experiments were made with a grayscale optical flow model. However, a multi-channel approach should provide better results, so we extended extended our original method into a multichannel scheme. Now, we evaluate the possible benefits of color against grayscale information in a variational optical flow method. The experiments are made with some synthetic sequences using RGB and grayscale images from the *Middlebury* benchmark database and the *MPI-Sintel* dataset.

Our interest is to compare motion details between the best flows achieved for the same sequence. We have made our evaluation using the best configuration of α and γ for every sequence. The remaining parameters are set as in the previous experiments.

The best flows (and its motion details) for the *Middlebury* sequences of *Grove3*, *RubberWhale* and *Urban2* are shown in figures 4.14, 4.15 and 4.16, respectively. On the other hand, the results for *Bandage 1*, *Bandage 2* and *Ambush 5* from *MPI-Sintel* dataset are depicted in the figures 4.17, 4.18 and 4.19.

Once again, we represent the motion with the IPOL color scheme, that appears in the first column of these figures. In the second column, we depict the true flow and the motion fields for grayscale and RGB, respectively. The remaining columns present motion details for the corresponding solutions.

The results are conclusive and confirm that the color information benefits the optical flow estimation. In general, the motion contours are better preserved and the error decreases perceptibly, especially in figures 4.14 and 4.19.

Figure 4.14 is a good example of this improvement, where the solution achieved by the color image is closer to the true flow than its grayscale counterpart. If we observe the motion details, we can see a more realistic solution that achieves a better preservation of the leafs. The reasons behind this enhancement is that the difference between the leafs and the background is much pronounced in the color image. In this sense, the grayscale picture makes it hard for the method to distinguish the moving objects with respect to the background.

For the same reasons, the definition of the contour is more accurate in figures 4.15 and 4.16. For instance, we observe an improvement in the details exposed in the red box of the color solution of *RubberWhale*. The multi-channel information helps in the the shading region. Nevertheless, the benefits are not so clear in both figures as the previous one.

The flows of figure 4.17 are more precise in its third row. Here, the diffusion in the dragon tail (green box) is ameliorated due to the color information. On the other hand, we can also see a significant decrease of the error in the head of the dragon at figure 4.18.

Finally, although neither of the motion fields of figure 4.19 offer good results, we observe fewer outliers in the color solution.

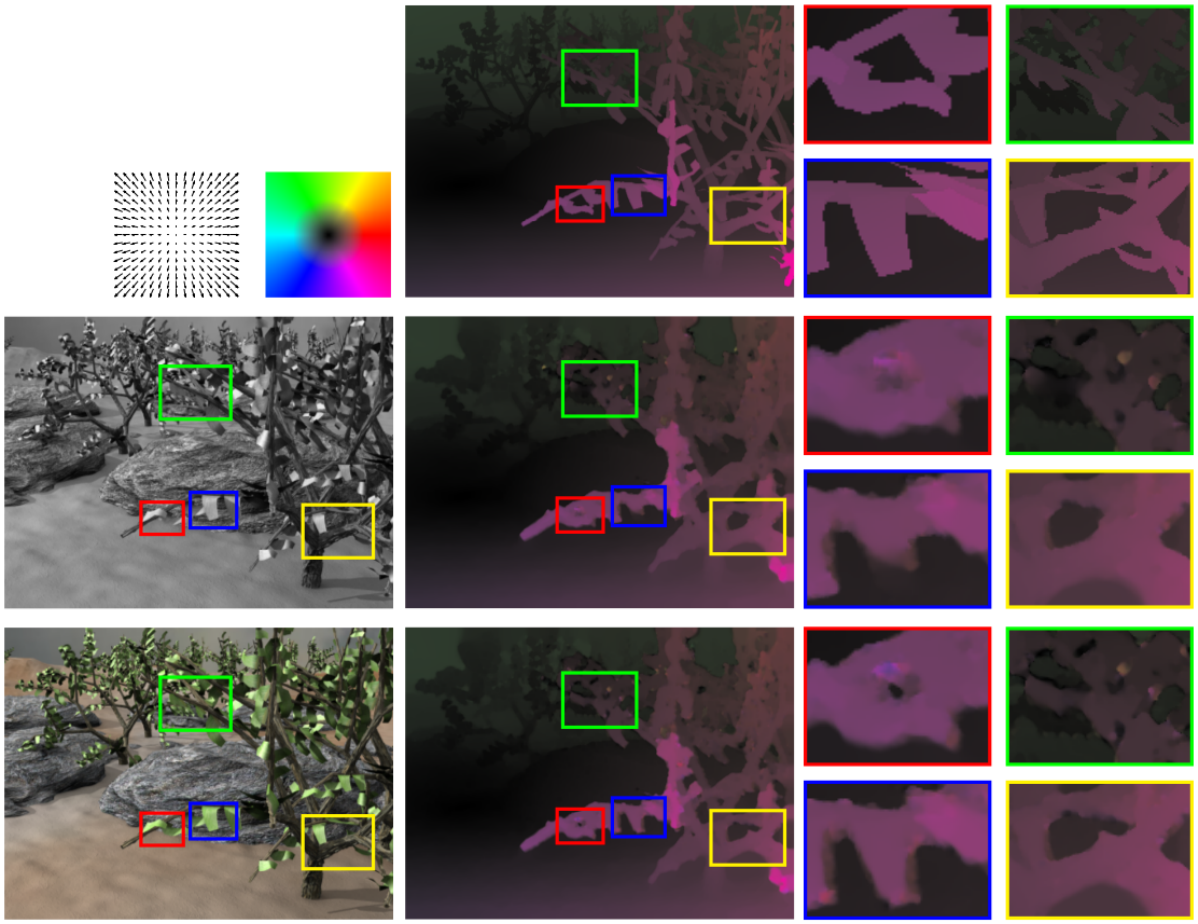


Figure 4.14: Motion details for the *Grove3* sequence. First row depicts the color scheme, the true flow and its details at the first, second and third columns, respectively. Second row shows the original grayscale image, the flow field achieved and some details for the comparison. Third row shows the results for the color image. According to the results, we see that the leafs are more defined in the color solutions.

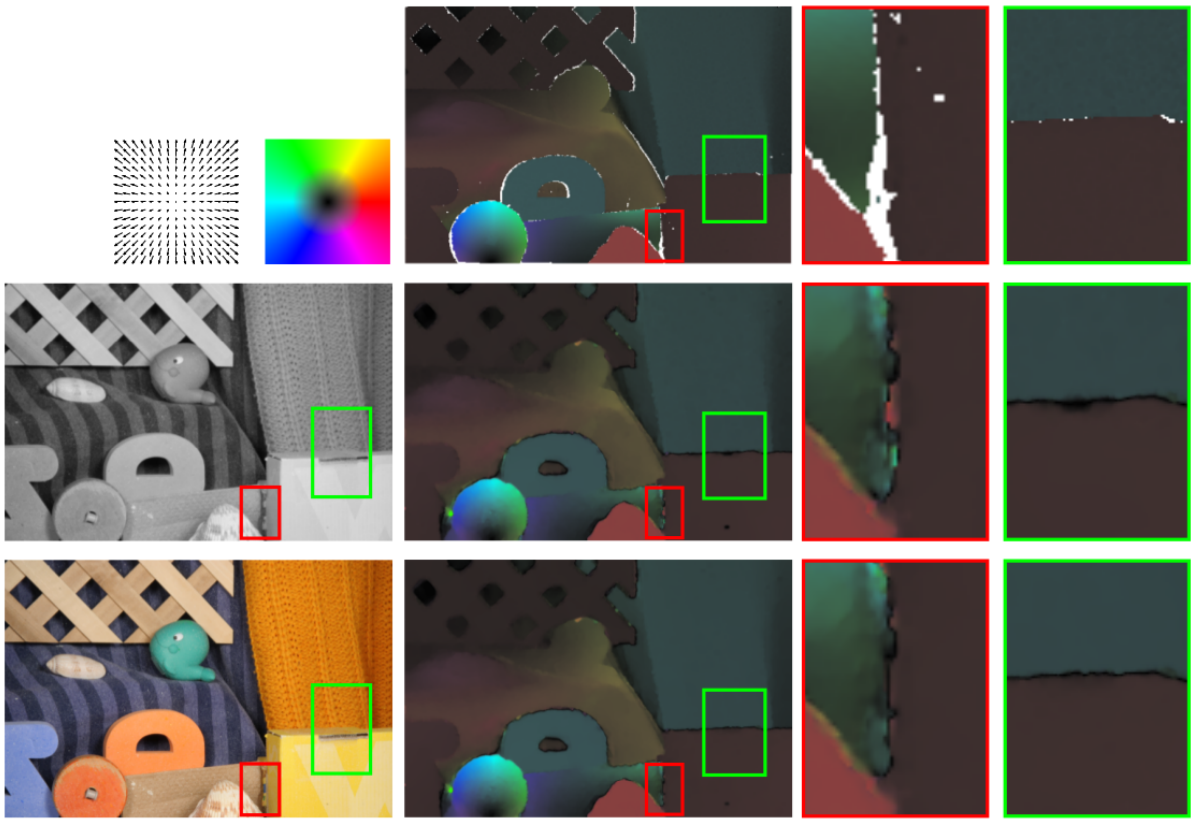


Figure 4.15: Motion details for the *RubberWhale* sequence. First row depicts the color scheme, the true flow and its details at the first, second and third columns, respectively. Second row shows the original grayscale image, the flow field achieved and some details for the comparison. Third row shows the results for the color image. The solutions seems to be similar but, the errors decrease if we observe the motion details.

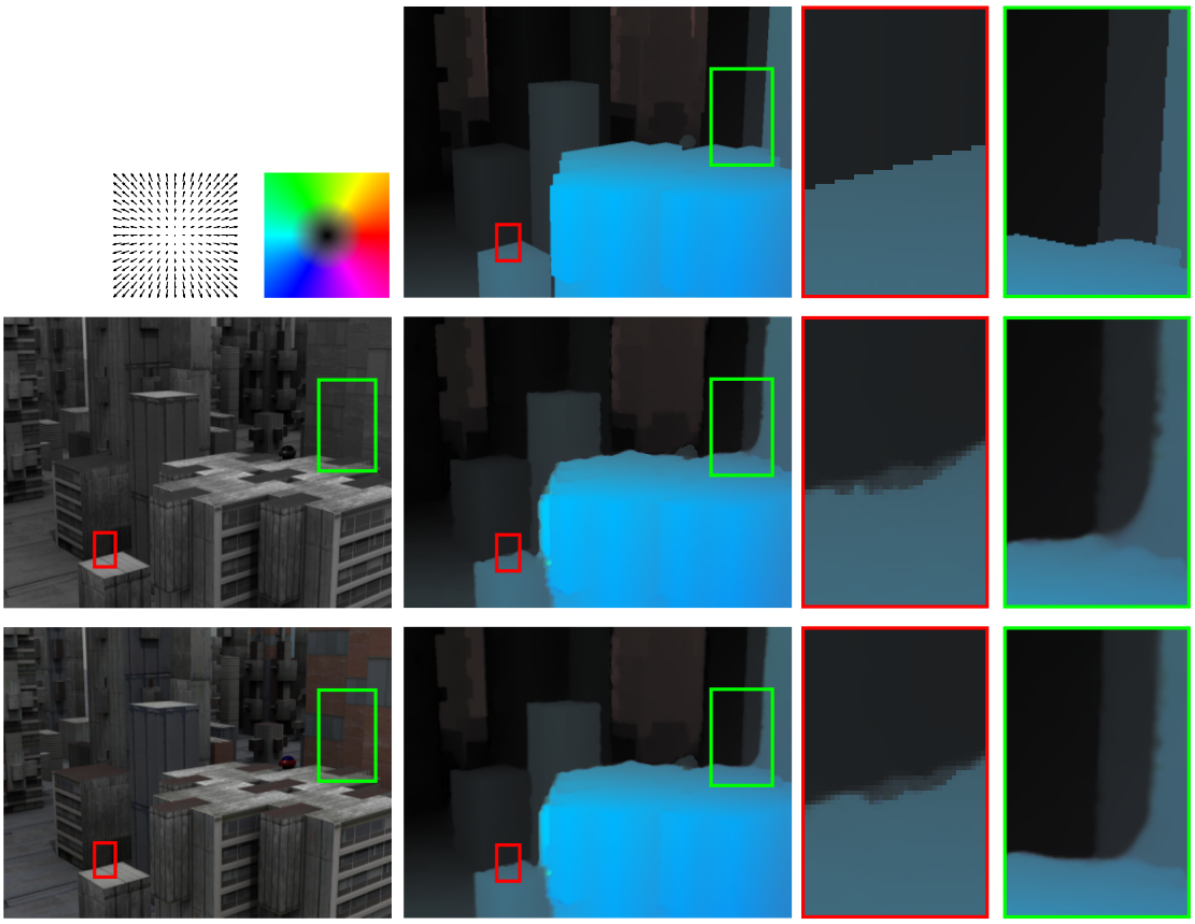


Figure 4.16: Motion details for the *Urban2* sequence. First row depicts the color scheme, the true flow and its details at the first, second and third columns, respectively. Second row shows the original grayscale image, the flow field achieved and some details for the comparison. Third row shows the results for the color image. The color information presents a better definition at the motion contours.

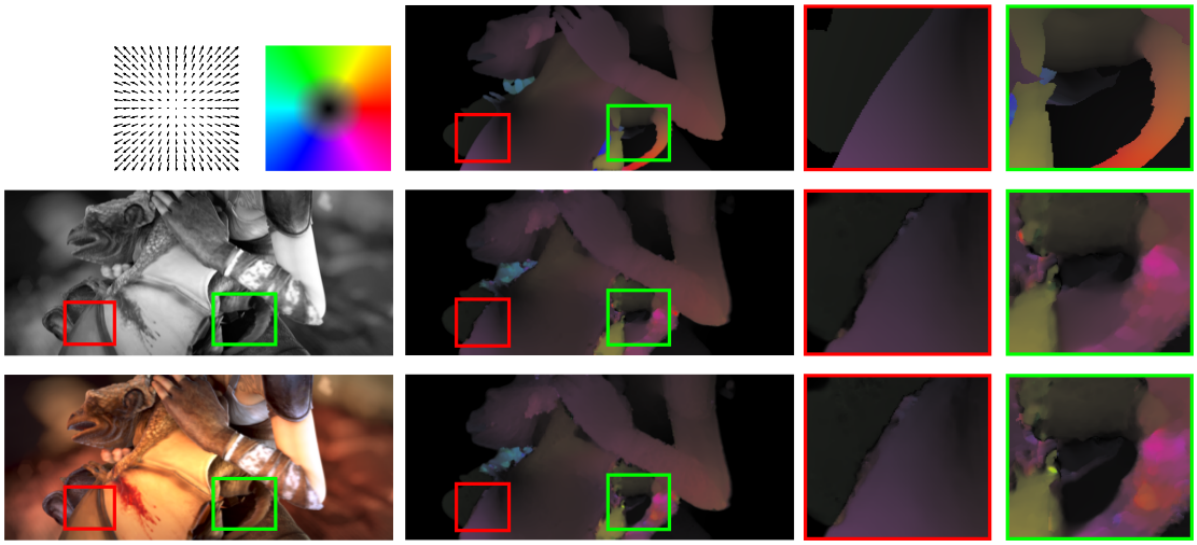


Figure 4.17: Motion details for the *Bandage 1*. First row depicts the color scheme, the true flow and its details at the first, second and third columns, respectively. Second row shows the original grayscale image, the flow field achieved and some details for the comparison. Third row shows the results for the color image. The motions details reveal an enhancement in the contours.

Figure 4.18: Motion details for *Bandage 2*. First row depicts the color scheme, the true flow and its details at the first, second and third columns, respectively. Second row shows the original grayscale image, the flow field achieved and some details for the comparison. Third row shows the results for the color image. The dragon head is closer to the true flow than the grayscale solution.

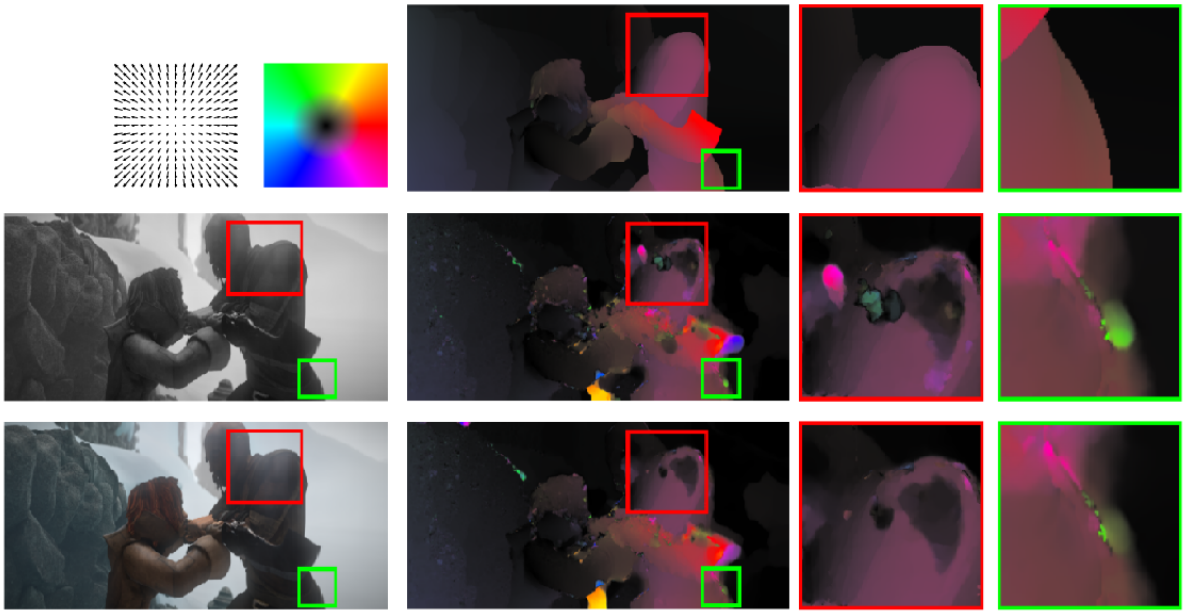


Figure 4.19: Motion details for *Ambush 5*. First row depicts the color scheme, the true flow and its details at the first, second and third columns, respectively. Second row shows the original grayscale image, the flow field achieved and some details for the comparison. Third row shows the results for the color image. In both results, the solutions are not accurate. However, the color solution is better that the grayscale flow.

4.3 Numerical Results

In this section, we compare the numerical results (AAE and EPE) provided by the best configurations of α and γ using color or grayscale in these two datasets. Furthermore, the computational cost is very important if we need to choose the best scheme; therefore, we take into account the running times required by the methods to justify if the computational cost its reasonable. The values for the remaining parameters have been fixed as in the previous section.

Tables 4.5 and 4.6 show the best AAE and the EPE results for both color spaces while, its fourth and seventh columns, depict the percentage of error variation between them. Once again, it is clear that the RGB information improves the numerical errors, especially in the *Middlebury* sequences where most of the amelioration in the EPE exceeds in 5.50% and the AAE in 4%.

Moreover, the enhancement is not so evident in the *MPI Sintel* sequences. It can be appreciated that, in a few occasions, the color worsens the EPE. Nevertheless, this deterioration is not excessive and the color information provides a reasonable improvement in a high percentage of cases, especially in *Ambush 5* and *Sleeping 1*.

The error amelioration is not enough to determine if a multi-channel scheme is justified. Thus, Figures 4.7 and 4.8 show the execution times needed for obtaining all the previous results. The size of the sequences from MPI Sintel is 1024×436 while, in the case of *Middlebury*, the size of the images is variable: *Grove2*, *Grove3*, *Urban2* and *Urban3* are 640×480, while *Hydrangea*, *RubberWhale* and *Dimetrodon* are 584×388.

Tables 4.7 and 4.8 show the running times of each sequence and the percentage of improvement provided by the grayscale with respect to the RGB images. As expected, less information means less execution time. However, the difference is not excessively pronounced and, even, in the case of *Ambush 2* sequence, less time is required if we use color information. This means that, using more information from the images, the algorithm usually converges in less iterations. Moreover, we must consider that the increment in the calculations has only affected the data term, so that more image channels does not mean that the operations grow up proportionally. From these experiments, we conclude that if our main goal resides in the precision of the results, the RGB information is quite interesting but, if the performance is required, a grayscale method is an alternative to consider. Besides, we can parallelize the multi-channel framework or use other numerical schemes such as multigrid [Bruhn06].

The experiments were realized in a computer with the following features: Intel(R) Core(TM) i7 CPU 860 @2.80GHz. We have used a single core in our tests.

Table 4.5: AAE and EPE for the *Middlebury* dataset.

	AAE			**EPE**		
Sequence	**Grayscale**	**RGB**	%	**Grayscale**	**RGB**	%
Grove2	2.198^o	2.169^o	1.34%	0.152	0.149	2.03%
Grove3	5.971^o	5.661^o	5.48%	0.659	0.612	7.68%
Hydrangea	2.142^o	2.015^o	6.30%	0.180	0.166	8.43%
RubberWhale	3.453^o	3.305^o	4.48%	0.103	0.097	6.18%
Urban2	2.438^o	2.328^o	4.72%	0.359	0.340	5.59%
Urban3	3.539^o	3.394^o	4.27%	0.386	0.401	-3.74%
Dimetrodon	1.588^o	1.458^o	8.91%	0.083	0.074	12.16%

Table 4.6: AAE and EPE for the MPI Sintel dataset.

	AAE			**EPE**		
Sequence	**Grayscale**	**RGB**	%	**Grayscale**	**RGB**	%
Alley 1	3.343^o	3.328^o	0.45%	0.374	0.373	0.27%
Alley 2	3.100^o	3.065^o	1.14%	0.270	0.269	0.37%
Ambush 2	31.83^o	31.12^o	2.28%	25.14	25.22	-0.32%
Ambush 4	28.24^o	27.92^o	1.14%	34.73	34.72	0.03%
Ambush 5	29.59^o	26.91^o	6.80%	2.647	2.234	18.5%
Ambush 6	12.21^o	11.95^o	2.17%	12.72	12.62	0.79%
Ambush 7	6.924^o	6.821^o	1.51%	2.072	2.152	-3.72%
Bamboo 1	4.416^o	4.237^o	4.22%	0.307	0.291	5.50%
Bamboo 2	6.334^o	6.084^o	4.10%	0.706	0.665	4.11%
Bandage 1	7.367^o	7.262^o	1.44%	1.061	1.030	3.01%
Bandage 2	9.403^o	9.242^o	1.74%	1.145	1.074	6.61%
Cave 2	3.219^o	3.166^o	1.67%	1.707	1.741	-1.95%
Cave 4	13.19^o	12.60^o	4.68%	7.810	7.681	1.68%
Market 2	8.814^o	8.628^o	2.15%	1.561	1.607	-2.86%
Market 5	33.68^o	32.84^o	2.55%	27.37	25.08	9.13%
Market 6	6.491^o	6.070^o	6.93%	9.245	9.036	2.31%
Mountain 1	6.500^o	6.296^o	3.24%	1.407	1.273	10.5%
Shaman 2	6.367^o	6.335^o	0.50%	0.311	0.307	1.30%
Shaman 3	6.105^o	6.038^o	1.11%	0.508	0.520	-2.31%
Sleeping 1	1.394^o	1.290^o	8.06%	0.113	0.104	8.65%
Sleeping 2	1.589^o	1.541^o	3.11%	0.070	0.067	4.47%
Temple 2	7.579^o	7.296^o	3.88%	1.438	1.456	-1.24%
Temple 3	3.165^o	2.991^o	5.82%	0.958	0.886	8.13%

Table 4.7: Runtime for the *Middlebury* sequences (in seconds).

	Runtime		
Sequence	**Grayscale**	**RGB**	**Speed-up** (%)
Grove2	45	67	32.83%
Grove3	49	72	31.94%
Hydrangea	22	46	52.17%
RubberWhale	30	48	37.50%
Urban2	50	72	30.55%
Urban3	55	81	32.10%
Dimetrodon	28	47	40.42%

Table 4.8: Runtime for the MPI Sintel dataset (in seconds).

	Runtime		
Sequence	**Grayscale**	**RGB**	**Speed-up** (%)
Alley 1	63	107	41.12%
Alley 2	49	81	39.51%
Ambush 2	199	150	-32.67%
Ambush 4	70	106	33.96%
Ambush 5	102	135	24.45%
Ambush 6	99	140	29.29%
Ambush 7	97	144	32.64%
Bamboo 1	64	116	44.83%
Bamboo 2	79	108	26.85%
Bandage 1	89	128	30.47%
Bandage 2	93	128	27.34%
Cave 2	84	113	25.66%
Cave 4	86	120	28.33%
Market 2	73	114	35.96%
Market 5	205	205	0.00%
Market 6	104	136	23.53%
Mountain 1	45	81	44.45%
Shaman 2	67	101	33.66%
Shaman 3	89	123	27.64%
Sleeping 1	31	63	50.79%
Sleeping 2	36	68	47.06%
Temple 2	92	131	29.77%
Temple 3	90	124	27.42%

4.4 Conclusion

In this chapter, we implemented the Spatial and Temporal approaches of the method proposed in [Brox04]. Both methods are very similar, with the main difference that the Temporal method performs a continuous smoothing scheme in the Temporal dimension. This is suitable if the optical flow functions are highly continuous. Nevertheless, the flow discontinuities are typically degraded by the Temporal regularizer. We observe that, in the presence of large discontinuities, it is better to use a nonlinear Temporal scheme, like in chapter 3.

The benefits of the TV-L^1 methods are important. They produce piecewise-smooth flow fields and they are robust against outliers. However, these kind of methods have several drawbacks: (i) they create rounded shapes near the borders or corners of the objects; (ii) typically, the edges usually do not coincide with the image contours. These two problems are due to the fact that the regularization process does not depend on the image information but on the flow field.

On the other hand, we studied the influence of color in variational optical flow methods. We developed a multi-channel implementation of the previous Spatial method. Our results confirm that: (i) the RGB color space offers several benefits in motion estimation without using new assumptions in the energy model and (ii) the computational cost does not grow up proportionally to the number of channels.

The multi-channel method decreases the number of outliers and also provides a better preservation of the discontinuities. In general, the color information allows to increase the accuracy, as we showed in the experiments. However, the improvement in many sequences from the *MPI-Sintel* database is not so evident. These sequences include many complex effects, such as fog, dust or blur that considerably increase the errors, independently of the image type. In fact, we observe that the color information ameliorates the solution in regions not affected by these perturbations.

The use of several channels suggests that the algorithm complexity will be higher and the computational cost will augment proportionally. Nevertheless, from the implementation point of view, the modifications introduced in the original model are straightforward. Even more, the runtime does not increase in line with the number of channels due to the following reasons: the changes only affect the data term and, typically, the color information allows a faster convergence of the algorithm. This cost can also be reduced if we use a multi-core infrastructure. We may conclude that the benefits of color information in variational optical flow methods compensate the increment on the execution time.

In the following chapter, we use a multi-channel scheme and introduce decreasing functions in order to stop the diffusion at image boundaries to avoid the inconveniences of TV-L^1 methods while maintaining its advantages.

Chapter 5

Robust Discontinuity-Preserving Optical Flow Methods

The preservation of motion discontinuities is still one of the main challenges in optical flow methods. The variational solutions are obtained as the minimization of a continuous functional where a smoothness term ensures that our optical flow is piece-wise continuous. In the other hand, it is critical for the correct preservation of motion boundaries. This is an important problem due to the fact that a poor preservation of flow edges makes difficult to separate different moving regions.

These boundaries are normally associated with the contours of the objects. Nevertheless, an object border does not necessarily imply a flow boundary. For example, it is not strange that a scene presents adjacent objects that are moving in the same direction. In this case, these ones probably belong to the same optical flow region without motion discontinuities between them. Besides, it is challenging to differentiate between object contours and textures.

Many works have proposed alternatives in order to cope with motion discontinuities. Several optical flow methods [Proesmans94, Black93, Cohen93, Brox04, Zach07] use anisotropic diffusion to reduce the effect of outliers and produce sharp boundaries. In other works [Nagel86, Alvarez00], the regularization process is steered by a diffusion tensor that depends on the image structures.

This idea has been used in more recent methods like [Sun08, Werlberger09]. The difference is that they include robust function in order to avoid oversegmenting the flow field. In [Zimmer09, Zimmer11], the authors include also a motion tensor in the data term. Other smoothing strategies include bilateral filtering [Xiao06] or non-local regularizations, like in [Werlberger10].

As we have seen in chapter 4, TV-L^1 methods produce piecewise smooth motion fields at the same time that they deal with outliers and textured areas. Nevertheless, they also

Figure 5.1: Instabilities problem. Top row: first and second frames of a sequence of a moving taxi. Bottom row: first column, the solution obtained with the Brox *et al.* method; second column, the optical flow obtained with a discontinuity-preserving strategy (note the instabilities that appear in some parts of the motion field); third column, the solution obtained with a similar technique that deals with this type of instabilities.

tend to create rounded shapes near the borders of the objects. This problem can be solved with a decreasing function that inhibits the smoothing in the areas where the gradient of the is strong [Alvarez99].

This strategy is simple and offers interesting results in the literature [Wedel09a, Xu10]. In fact, it is probably the most used alternative. However, it is especially difficult to determine the correct parameters for the decreasing function. This is problematic due to the fact that if the parameter are underestimated, the solutions do not improve with respect to a basic TV approach while, on the other hand, if it is overestimated it is quite easy that it becomes an ill-posed problem and may introduce some instabilities [Monzón13]. Figure 5.1 shows an example of these situations using the *Taxi* sequence. In the second row, from left to right, we observe the mentioned rounded shapes that usually appear in the Brox method, the instabilities in the form of blobs that we may find in a typical discontinuity-preserving method and a correct solution that we obtain from the method that will be explained in the following.

In our experiments, we notice that this is an important challenge, which is not anecdotal and usually appears in most of the sequences. In fact, it is quite difficult to establish a common parameter setting for several types of sequences.

Therefore, we analyze several discontinuity-preserving techniques and the influence of

the parameters. In particular, we focus on study in the combination of robust functions and diffusion tensors, including the use of exponential functions. Our study has led us to propose two efficient strategies to overcome this problem. These approaches reduce the effect of instabilities and the probability of failures in the selection of correct parameters for a a broad variety of sequences.

Our first approach ensures a minimum isotropic regularization that prevents the cancellation of the smoothness term by using a small constant. A similar idea was introduced in [Ayvaci12] but, in that case, the regularization was carried out for each component of the optical flow independently. This strategy allows a better performance at flow discontinuities and it is more insensitive to the discontinuity parameter. Nevertheless, user intervention is still necessary for choosing it.

In this sense, our second proposal estimates its value from the gradient of the image and the regularization parameter. This means that the method is automatically adapted to the image range at the same time that it provides good motion contours. In the experiments, we observe that it usually obtains accurate flow fields which are close to the best solutions, without any user intervention.

We also analyze a scheme similar to the robust diffusion tensors proposed in [Werlberger09] or [Zimmer11]. The diffusion across image contours is controlled with a robust decreasing function and maintained along the isocontours of the objects. In this case, the method is very stable with respect to the anisotropic parameter. In the experiments, we show the capabilities of these strategies and the benefits and drawbacks with respect to a standard TV-L^1 method.

The chapter is organized as follows: in Section. 5.1, we introduce a general framework for the regularization strategies and we explain each proposal in detail. In Section 5.2, we show our experimental results. First, we analyze each strategy and their behavior with respect to the instabilities problem; then, we compare all the strategies together. Finally, a summary of the main ideas and conclusion are given in Section 5.5.

5.1 Energy Functional

Let $I_1^c, I_2^c : \Omega \subset \mathbb{R}^2 \rightarrow \mathbb{R}^c$ be a sequence of images, with $\mathbf{x} = (x, y)^T \in \Omega$, $\{I^c\}_{c=1,\dots,C}$ and C the number of channels. $\mathbf{w}(\mathbf{x}) = (u(\mathbf{x}), v(\mathbf{x}))^T$ defines the optical flow between two consecutive images. Functions $u(\mathbf{x})$ and $v(\mathbf{x})$ are the horizontal and vertical displacements, respectively.

According to this notation, our energy functional reads as

$$E(\mathbf{w}) = \int_{\Omega} \mathcal{D}\left(I_1, I_2, \nabla I_1, \nabla I_2, \mathbf{w}\right) \mathbf{dx} + \alpha \int_{\Omega} \mathcal{R}\left(\nabla I_1, \nabla u, \nabla v\right) \mathbf{dx}, \tag{5.1}$$

Table 5.1: List of the regularizers analyzed in this work and the corresponding diffusion tensors, $\mathbf{D}$. $\mathbf{I}$ stands for the identity matrix, $\mathbf{n} = \frac{\nabla I_1}{|\nabla I_1|}$ and $\mathbf{n}^{\perp}$ a normalized vector in the direction of the gradient and $\Psi(s^2) = \sqrt{s^2 + \epsilon^2}$, $\Psi'(s^2) = \frac{1}{2\sqrt{s^2+\epsilon^2}}$, $\Phi(s^2) = \frac{1}{\lambda^2} log\left(1 + \lambda^2 s^2\right)$ and $\Phi'(s^2) = \frac{1}{1+\lambda^2 \cdot s^2}$.

Method	$\mathcal{Z}$	$\mathcal{R}\left(\nabla I_1, \nabla u, \nabla v\right)$	$\mathbf{D}\left(\nabla I_1, \nabla u, \nabla v\right)$
Brox	1	$\Psi\left(\mathcal{Z} \cdot \left(\lvert\nabla u\rvert^2 + \lvert\nabla v\rvert^2\right)\right)$	$\mathcal{Z} \cdot \Psi'\left(\mathcal{Z} \cdot \left(\lvert\nabla u\rvert^2 + \lvert\nabla v\rvert^2\right)\right) \cdot \mathbf{I}$
DF	$e^{-\lambda\lvert\nabla I_1\rvert}$		
DF-β	$e^{-\lambda\lvert\nabla I_1\rvert} + \beta$		
DF-Auto	$e^{-\lambda_\pi\lvert\nabla I_1\rvert}$		
RADT	-	$\Phi\left(\left(\mathbf{n}^T\nabla u\right)^2 + \left(\mathbf{n}^T\nabla v\right)^2\right)$ $+\left(\mathbf{n}^{\perp}\nabla u\right)^2 + \left(\mathbf{n}^{\perp}\nabla v\right)^2$	$\Phi'((\mathbf{n}^T\nabla u)^2 + (\mathbf{n}^T\nabla v)^2)$ $+\mathbf{n}^{\perp}\mathbf{n}^{\perp T}$

being $\mathcal{D}$ and $\mathcal{R}$, the data and the regularization terms, respectively.

As in the original multi-channel scheme, we assume that the brightness and gradient constancy assumptions are also fulfilled in our framework. Our data term is given by

$$\mathcal{D}\left(I_1, I_2, \nabla I_1, \nabla I_2, \mathbf{w}\right) := \Psi\left(\sum_{c=1}^{C}\left(I_2^c(\mathbf{x}+\mathbf{w}) - I_1^c(\mathbf{x})\right)^2\right) + \gamma\Psi\left(\sum_{c=1}^{C}\left|\nabla I_2^c(\mathbf{x}+\mathbf{w}) - \nabla I_1^c(\mathbf{x})\right|^2\right), \tag{5.2}$$

with $\Psi(s^2) = \sqrt{s^2 + \epsilon^2}$ and $\epsilon := 0.001$.

We use the regularization terms presented in table 5.1: the Brox *et al.* [Brox04] method, a robust anisotropic diffusion tensor (in the same way as [Werlberger09]) and three variants of $\mathcal{R}\left(\nabla I_1, \nabla u, \nabla v\right) := \Psi\left(g(|\nabla I_1|) \cdot \left(|\nabla u|^2 + |\nabla v|^2\right)\right)$. The decreasing function that inhibits regularization at object contours is represented by $g(\cdot)$. Some typical alternatives are

$$g\left(\nabla I_1\right) = e^{-\lambda|\nabla I_1|^{\kappa}},\ g\left(\nabla I_1\right) = \frac{1}{1 + \lambda\left|\nabla I_1\right|^2}. \tag{5.3}$$

We use $\kappa := 1$ in our experiments. This value has been empirically determined as a compromised between stability and accuracy, as explained in section 5.2.1.

Table 5.2 shows some well known regularisers in the literature, their formulation and their main features. In Zimmer *et al.*. [Zimmer11], the regulariser is decomposed in

eigenvalues (μ_1, μ_2) and eigenvectors (r_1, r_2) of the following structure tensor: $\mathbf{R}_\rho := \sum_{c=1}^{3} K_\rho * \left[\theta_0^c \left(\nabla_2 I^c \nabla_2^T I^c\right) + \gamma \left(\theta_x^c \left(\nabla_2 I_x^c \nabla_2^T I_x^c\right) + \theta_y^c \left(\nabla_2 I_y^c \nabla_2^T I_y^c\right)\right)\right]$.

5.1.1 Regularization Strategies

Brox approach

$$\mathcal{R}\left(\nabla u, \nabla v\right) := \Psi\left(|\nabla u|^2 + |\nabla v|^2\right).$$

The Brox method has been deeply analyzed in the chapter 4. In the present chapter we compare its behavior with other schemes. In this sense, we have to remember that the Brox method proposes a robustification of the Horn-Schunck regulariser. The Total Variation function creates piecewise and continuous motion regions, but, at the same time, it creates rounded and dislocated contours.

In Fig. 5.3 we can see the effect of robustification and how the diffusion process is cut off with respect to the Horn-Schunck flow field.

Regularization with Decreasing Scalar Function (*DF*)

$$\mathcal{R}\left(\nabla I_1, \nabla u, \nabla v\right) := \Psi\left(e^{-\lambda|\nabla I_1|} \cdot \left(|\nabla u|^2 + |\nabla v|^2\right)\right).$$

The *DF* method includes a decreasing function to avoid smoothing at flow discontinuities. We see that it effectively reduces the rounded effects at flow edges and obtain more continuous flows. This is due to the fact that the decreasing functions disable the smoothing at large image gradients. However, the parameters must be chosen carefully in order to avoid a degradation of the motion fields.

Regularization with Decreasing Scalar Function and Constant Diffusion (*DF-β*)

$$\mathcal{R}\left(\nabla I_1, \nabla u, \nabla v\right) := \Psi\left(\left(e^{-\lambda|\nabla I_1|} + \beta\right) \cdot \left(|\nabla u|^2 + |\nabla v|^2\right)\right).$$

The introduction of a small constant, β, ensures a minimum isotropic diffusion, which reduces the effect of instabilities. This constant must be big enough for ensuring that there always exists a smoothing process, but, it must be also sufficiently small to preserve the good features of the former approach. In the experiments we carry out a study to find a suitable value for β. We conclude that $\beta := 0.0001$ is this value.

Table 5.2: Summary of some regularization strategies proposed in the literature. $\mathbf{I}$, the identity matrix. $g(\cdot)$ a decreasing function. $\mathbf{n} = \frac{\nabla I_1}{|\nabla I_1|}$ and $\mathbf{n}^\perp$ its orthonormal vector. R is the robust function that can be applied to *'all'* or only $\mathbf{n}, \mathbf{n}^\perp$ directions. $\mathcal{R}$ is the regulariser.

Rotationally Invariant			
Method	R.	Norm	$\mathcal{R}(\nabla I_1, \nabla u, \nabla v)$
Horn-Schunck [Horn81]	-	L^2	$\lvert\nabla u\rvert^2 + \lvert\nabla v\rvert^2$
Alvarez *et al.* [Alvarez99]	-	L^2	$g(\lvert\nabla I_1\rvert) \cdot (\lvert\nabla u\rvert^2 + \lvert\nabla v\rvert^2)$
Brox *et al.* [Brox04]	all	L^1	$\Psi\left(\lvert\nabla u\rvert^2 + \lvert\nabla v\rvert^2\right)$
Monzón *et al.* [Monzón14a]	all	L^1	$\Psi(g(\lvert\nabla I_1\rvert) + \beta) \cdot (\lvert\nabla u\rvert^2 + \lvert\nabla v\rvert^2))$
Nagel-Enkelmann [Nagel86]	-	L^2	$\nabla u^T \mathcal{Z} \nabla u + \nabla v^T \mathcal{Z} \nabla v$, $\mathcal{Z} = \frac{\nabla I_1^\perp \nabla I_1^{\perp T} + \beta^2 \mathbf{I}}{\lvert\nabla I_1\rvert^2 + 2\beta^2}$
Sánchez *et al.* [Sánchez13f]	all	L^1	$\Psi\left(\nabla u^T \mathcal{Z} \nabla u + \nabla v^T \mathcal{Z} \nabla v\right)$
Zimmer *et al.* [Zimmer11]	$\mathbf{n}$	L^1	$\Psi\left(u_{r_1}^2 + v_{r_1}^2\right) + (u_{r_2}^2 + v_{r_2}^2)$, (r_1, r_2) eigenvalues of a structure tensor.
Non-Rotationally Invariant			
Method	R.	Norm	$\mathcal{R}(\nabla I_1, \nabla u, \nabla v)$
Zach *et al.* [Zach07]	-	L^1	$\lvert\nabla u\rvert + \lvert\nabla v\rvert$
Xu *et al.* [Xu10]	all	L^1	$g(\lvert\nabla I_1\rvert) \cdot (\lvert\nabla u\rvert + \lvert\nabla v\rvert)$
Werlberger *et al.* [Werlberger09]	all	Huber-L^1	$\left\lvert\left(g(\lvert\nabla I_1\rvert^2)\mathbf{n}\mathbf{n}^T + \mathbf{n}^\perp\mathbf{n}^{\perp T}\right)\nabla u\right\rvert_\epsilon + \left\lvert\left(g(\lvert\nabla I_1\rvert^2)\mathbf{n}\mathbf{n}^T + \mathbf{n}^\perp\mathbf{n}^{\perp T}\right)\nabla v\right\rvert_\epsilon$
Cohen [Cohen93]	all	L^1	$\lvert\nabla u\rvert + \lvert\nabla v\rvert$
Wedel *et al.* [Wedel09a]	all	L^1	$g(\lvert\nabla I_1\rvert) \cdot (\lvert\nabla u\rvert + \lvert\nabla v\rvert)$
Ayvaci *et al.* [Ayvaci12]	all	L^1	$(g(I_{1,x}) + \beta, g(I_{1,y}) + \beta) \cdot (\lvert\nabla u\rvert + \lvert\nabla v\rvert)$

Regularization with Automatic Setup of Decreasing Scalar Function (*DF-Auto*)

$$\mathcal{R}\left(\nabla I_1, \nabla u, \nabla v\right) := \Psi\left(e^{-\lambda_\pi |\nabla I_1|} \cdot \left(|\nabla u|^2 + |\nabla v|^2\right)\right).$$

The *DF-β* method improves the parameter selection when using decreasing functions. Nevertheless, the user still needs to find the appropriate value that solves the discontinuity problem. The *DF-Auto* strategy offers an interesting improvement, since the discontinuity parameter, λ_π, is estimated automatically. The idea is to adapt its value in order to avoid any possible instability.

The minimization of (5.1), with the regularisers proposed in this work, provides a solution in the form

$$\begin{aligned} 0 =& \partial_u \mathcal{D} - \alpha \operatorname{div}\left(\Psi' \cdot \nabla u\right), \\ 0 =& \partial_v \mathcal{D} - \alpha \operatorname{div}\left(\Psi' \cdot \nabla v\right). \end{aligned}$$

The instabilities normally arise when the diffusivity approaches zero or, equivalently, $\alpha \cdot \Psi' \approx 0$. In this sense, one way to avoid the *ill-posedness* is to ensure the following condition

$$\alpha \cdot \Psi' = \frac{\alpha \cdot e^{-\lambda |\nabla I_1|}}{\sqrt{e^{-\lambda |\nabla I_1|} \cdot \left(|\nabla u|^2 + |\nabla v|^2\right) + \epsilon^2}} \geq \xi > 0,$$

with ξ a small constant to avoid the cancellation of the divergence term. In our experiments, we fix $\xi := 0.05$ and analyze the reasons behind this value.

Given that the numerator decreases much faster than the denominator and that the problems appear for large values of the gradient of the image, λ can be calculated at the beginning of the process by imposing the following constraint: $\alpha e^{-\lambda |\nabla I_1|} \geq \xi > 0$. Then, we deduce a value of λ for each position as

$$\lambda(\mathbf{x}) := \frac{-\ln(\xi) + \ln(\alpha)}{|\nabla I_1(\mathbf{x})|},$$

with $|\nabla I_1(\mathbf{x})| = \max_c \{|\nabla I_1^c(\mathbf{x})|\}$.

Note that the gradient of the image is a function of $\mathbf{x}$. When the gradient is close to zero, λ tends to ∞. Therefore, this calculation must discriminate whether a pixel belongs to an homogeneous region or not. We propose to calculate λ_π in each pixel as

$$\lambda_\pi := \min\{\lambda_\Omega, \lambda(\mathbf{x})\},$$

with

$$\lambda_\Omega := \frac{-\ln(\xi) + \ln(\alpha)}{f(\nabla I_1)}, \tag{5.4}$$

where $f(\nabla I_1)$ is a statistical function of the image gradients.

One possible alternative is

$$f_1(\nabla I_1) = \tau \cdot \max_{\mathbf{x}\in\Omega}\{|\nabla I_1(\mathbf{x})|\},$$

with $\tau \in (0, 1]$.

Another alternative is to rely on the histogram of the gradient as

$$f_2(\nabla I_1) = |\nabla I_1(\mathbf{x}')|,$$

then, we reformulated the expression as

$$\lambda_\Omega := \frac{-\ln(\xi) + \ln(\alpha)}{|\nabla I_1(\mathbf{x}')|}, \tag{5.5}$$

where $\mathbf{x}'$ is such that $|\nabla I_1(\mathbf{x}')|$ is the rank $\tau \times |\Omega|$ among the image gradients and $\tau \in (0, 1]$.

In both cases, τ determines the conservative behavior of λ_Ω. The first alternative, with $\tau := 1$, means that we use the maximum gradient of the whole image. In the experiments and the previous work in [Sánchez14] can be seen that it prevents the occurrence of instabilities, however, it restricts the detection of less prominent motion contours. Varying the value of τ allows to detect more discontinuities. Nevertheless, it is difficult to fix a good parameter for all types of sequences, as we can see on the left graphic in figure 5.2.

For the second alternative, we see on the right graphic in figure 5.2 that the errors decrease with respect to τ and the best results are around 0.9 for all the sequences. In this sense, a suitable value for τ is $\tau := 0.94$. Since this strategy is more robust we will use $f_2(\nabla I_1)$ in the experiments. Other approaches can be used, such as statistics based on the the average or median values of the image gradients.

We must note that the value of the regularization parameter α is influenced by the decreasing function and, therefore, by the λ parameter. Thus, the automatic estimation of λ offers an indirect benefit, which is a joint normalization with the smoothness weight. In this sense, λ is adapted to the α value: when it is small, the value of λ also be small. Even, if α is smaller than 1, the value of λ will become negative and the exponential will turn increasing, which avoids canceling the regularization term.

Robust Anisotropic Diffusion Tensor (*RADT*)

$$\mathcal{R}(\nabla I_1, \nabla u, \nabla v) := \Phi\left(\left(\mathbf{n}^T\nabla u\right)^2 + \left(\mathbf{n}^T\nabla v\right)^2\right) + \left(\mathbf{n}^\perp\nabla u\right)^2 + \left(\mathbf{n}^\perp\nabla v\right)^2$$

with $\Phi(s^2) = \frac{1}{\lambda^2} log\,(1 + \lambda^2 s^2)$.

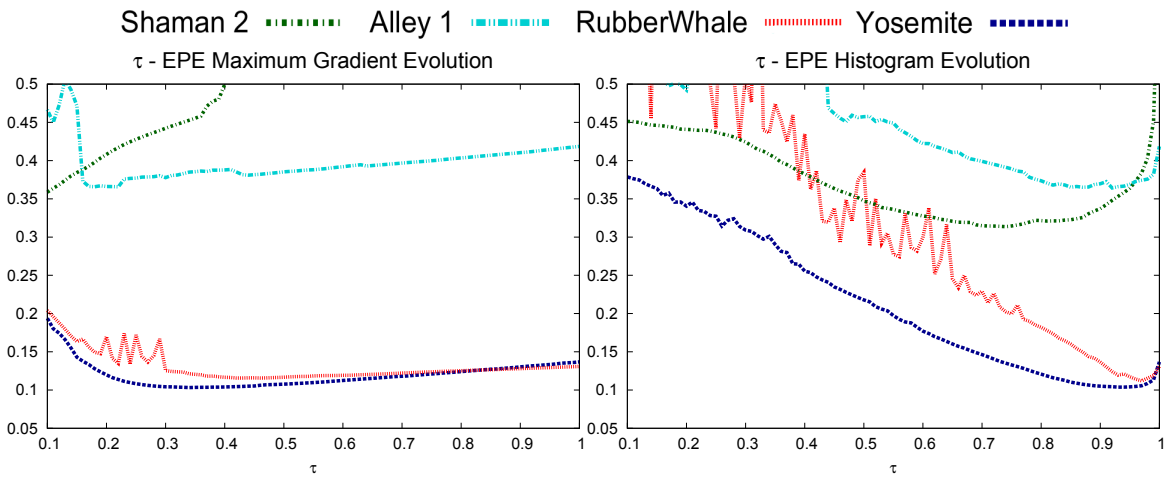


Figure 5.2: Average End Point Error (EPE) evolution with respect to τ using the two strategies proposed for λ estimation, $f_1(\nabla I_1)$ and $f_2(\nabla I_1)$. We see the evolution of the EPE error for several sequences. For the first proposal, we see that the results are very stable. However, the second proposal seems more interesting, because the best τ is bounded between 0.9 and 0.95.

This diffusion tensor resembles a robust variant of the Nagel-Enkelmann operator, with decreasing functions to mitigate the diffusion across the boundaries of the objects. The diffusion is performed unconditionally along the isocontours while in the gradient direction it is modulated by a robust function.

The behavior of the diffusion tensors is depicted in Fig. 5.3. We see how robustification techniques reduce the influence of noise and allow to obtain piecewise-smooth flow fields. The solution given by the Nagel-Enkelmann method improves the Horn-Schunck solution at the edges, as can be seen in the head of the girl.

Both, Brox and Nagel-Enkelmann methods, obtain piecewise-smooth motion fields. However, the Nagel-Enkelmann method cannot preserve discontinuities when the gradient of the image is small, as we see in the beard and the eyebrow. The Brox method detects both of them, although it does not distinguish the hair tufts. The Total Variation tends to smooth the flow destroying details. It also creates rounded shapes in high curvature boundaries.

The flow fields obtained by the Alvarez and *DF* methods are very similar. The most significant difference between them is the magnitude of the flow, especially in the eyebrow and beard. Furthermore, the robustification has allowed to detect discontinuities with small image gradients. In the *DF* approach, we see sharp discontinuities but small *artifacts* close to the old man ear.

The solution obtained by *RADT* identifies the main objects in the scene. However, it suffers the limitations of Nagel-Enkelmann and the motion edges are not so clear as in

the Brox or *DF* flow fields. The discontinuities are not clearly preserved due to the small gradients.

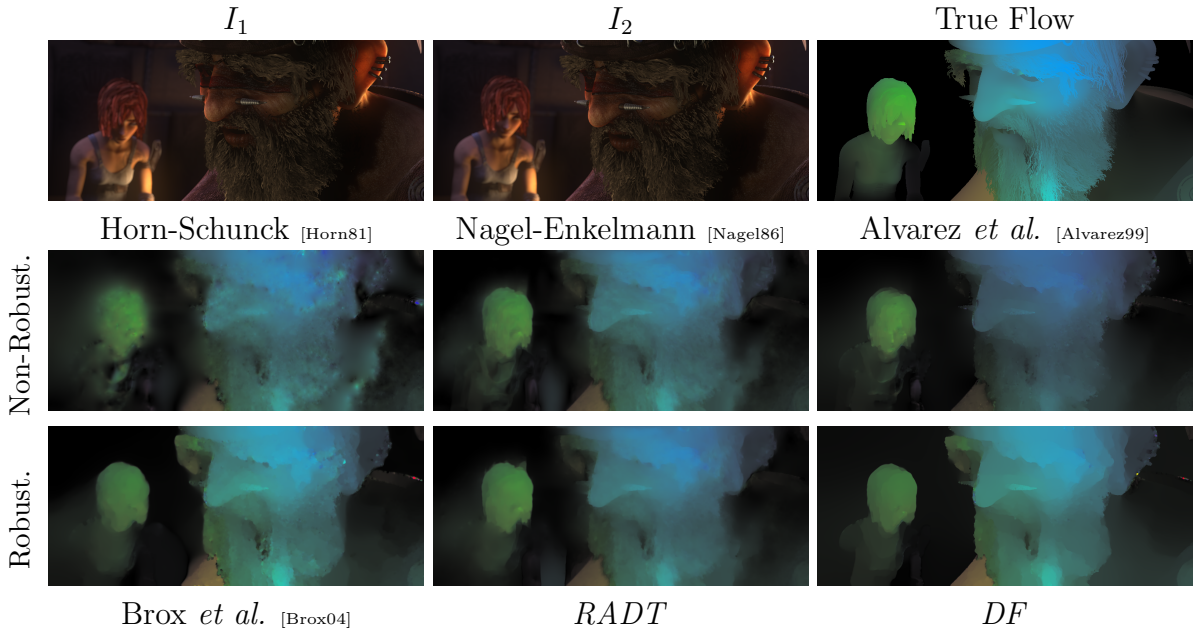


Figure 5.3: Top row: two frames of the *Shaman 2* sequence and its true flow. Middle row: the flow fields obtained with the Horn-Schunck, Nagel-Enkelmann and Alvarez methods. Bottom row: the solutions given by the Brox, a robust diffusion tensor and *DF* methods. We see how robust techniques reduce the influence of outliers and their combination with decreasing functions allows to obtain piecewise-smooth flow fields.

The minimum of the energy functional (5.1) can be found by solving the associated Euler-Lagrange equations. The data term is given in (5.2) and the smoothing term is adapted to the diffusion tensors presented in table 5.1. If we minimize (5.1), we obtain the following system of equations:

$$\begin{aligned} 0 = \Psi'_B \cdot \left(\sum_{c=1}^{C} I_2^c(\mathbf{x}+\mathbf{w}) - I_1^c(\mathbf{x}) \right) \cdot I_{2,x}^c(\mathbf{x}+\mathbf{w}) \\ + \gamma\, \Psi'_G \cdot \left(\sum_{c=1}^{C} \left(I_{2,x}^c(\mathbf{x}+\mathbf{w}) - I_{1,x}^c(\mathbf{x}) \right) \cdot I_{2,xx}^c(\mathbf{x}+\mathbf{w}) \right. \\ \left. + \sum_{c=1}^{C} \left(I_{2,y}^c(\mathbf{x}+\mathbf{w}) - I_{1,y}^c(\mathbf{x}) \right) \cdot I_{2,xy}^c(\mathbf{x}+\mathbf{w}) \right) \\ - \alpha \operatorname{div} \left(\mathbf{D} \left(\nabla I_1, \nabla u, \nabla v \right) \cdot \nabla u \right), \end{aligned}$$

$$
\begin{aligned}
0 = \Psi'_B \cdot & \left(\sum_{c=1}^{C} I_2^c(\mathbf{x}+\mathbf{w}) - I_1^c(\mathbf{x}) \right) \cdot I_{2,y}^c(\mathbf{x}+\mathbf{w}) \\
& + \gamma\, \Psi'_G \cdot \left(\sum_{c=1}^{C} \left(I_{2,x}^c(\mathbf{x}+\mathbf{w}) - I_{1,x}^c(\mathbf{x}) \right) \cdot I_{2,xy}^c(\mathbf{x}+\mathbf{w}) \right. \\
& \left. + \sum_{c=1}^{C} \left(I_{2,y}^c(\mathbf{x}+\mathbf{w}) - I_{1,y}^c(\mathbf{x}) \right) \cdot I_{2,yy}^c(\mathbf{x}+\mathbf{w}) \right) \\
& - \alpha\, \mathrm{div}(\mathbf{D}\left(\nabla I_1, \nabla u, \nabla v\right) \cdot \nabla v),
\end{aligned} \tag{5.6}
$$

with $\Psi'(s^2) = \frac{1}{2\sqrt{s^2+\epsilon^2}}$. In order to simplify these equations, we have used the following notation:

$$
\begin{aligned}
\Psi'_B :=& \Psi' \left(\sum_{c=1}^{C} \left(I_2^c(\mathbf{x}+\mathbf{w}) - I_1^c(\mathbf{x}) \right)^2 \right), \\
\Psi'_G :=& \Psi' \left(\sum_{c=1}^{C} \left| \nabla I_2^c(\mathbf{x}+\mathbf{w}) - \nabla I_1^c(\mathbf{x}) \right|^2 \right),
\end{aligned} \tag{5.7}
$$

and $\mathbf{D}\left(\nabla I_1, \nabla u, \nabla v\right)$ the diffusion tensors shown in Table 5.1.

We discretize the equations using centered finite differences and solve the system by means of the iterative SOR method. Due to the nonlinear nature of these formulas, the resolution of these equations requires two fixed point iterations, in order to converge to a steady state. The warping of I_2 are approximated using Taylor expansions.

These equations are embedded in a multi-scale strategy that allows to recover large displacements. Starting from the coarsest scales, we obtain a solution to the above system, and then upgrade the value of the optical flow for the next finer scale. We use *motion increments*, $\mathbf{w}^{m+1} = \mathbf{w}^m + \mathbf{dw}^m$, so that, in each scale, we compute each increment, $\mathbf{dw}^m$, and the final optical flow is obtained as an accumulative value for all increments. Details on the discretization of this scheme are given extensively in chapter 4.

5.2 Experimental Analysis

In this section, we analyze separately the different proposals for discontinuity-preserving optical flow methods using some standard image sequences. In particular, we use a set of images with geometric figures (a square, a star and two rectangles), presenting pronounced discontinuities and large displacements and the test and evaluation sequences from the *MPI Sintel* [Butler12] and *Middlebury* [Baker07b] benchmark databases, which allows us

to show the results in a more general setting. The optical flows are represented using the IPOL color scheme.

The aim of the experiments is to analyze the influence of the regularization in the preservation of discontinuities. Therefore, our study will basically concentrate on the α and λ parameters. In the majority of the experiments, γ has been set to zero. The remaining parameters have been fixed as in chapter 4.

5.2.1 Analysis of the *DF* Method

Fixing the value of the κ parameter

In previous works like in [Xu10] or [Wedel09a], the exponential function was defined in terms of two parameters:

$$g\left(\nabla I_1\right) = e^{-\lambda|\nabla I_1|^{\kappa}}. \tag{5.8}$$

Obviously, this means that the probability of a wrong configuration of the exponential function is bigger. Therefore, we compare the evolution of the Average Angular Error (AAE) in figure 5.4 and the Average End-Point Error (EPE) in figure 5.5 for the *Yosemite*, *Hydrangea*, *Grove3*, *Urban2* and *Urban3* sequences with respect to the λ parameter and for three values of κ.

In the case of *Yosemite*, the results are much more stable with respect to λ and κ than the other graphics. On the other hand, the graphics for *Urban2* and *Urban3* are more unstable, which is probably due to the small values of α. We observe that, typically, when λ increases, the AAE improves and the method provides better results, even for higher values of the smoothness parameter. In the EPE graphics, the error evolution is quite similar. As expected, when κ increases, the value of λ must decrease to achieve better solutions. At the light of these results, we fix $\kappa := 1$ in all the variants of the decreasing function in order to reduce their complexity.

Analysis of the λ parameter

Next, we use figures 5.6 and 5.7 for showing the influence of λ over the motion field with a constant α. As expected, we observe that a small λ does not completely stop the diffusion at contours and underestimates the magnitude of the optical flows.

However, the use of a large α and a correct λ produces that the resulting solutions are accurate and the discontinuities are correctly preserved, specially in the geometric figures. Besides, we observe that the exponential allows to fill the occluded regions with the value from the background. This is achieved because, on the one hand, it overweights the regularization with respect to the data term and, on the other hand, it does not mix the information with the flow inside the figures.

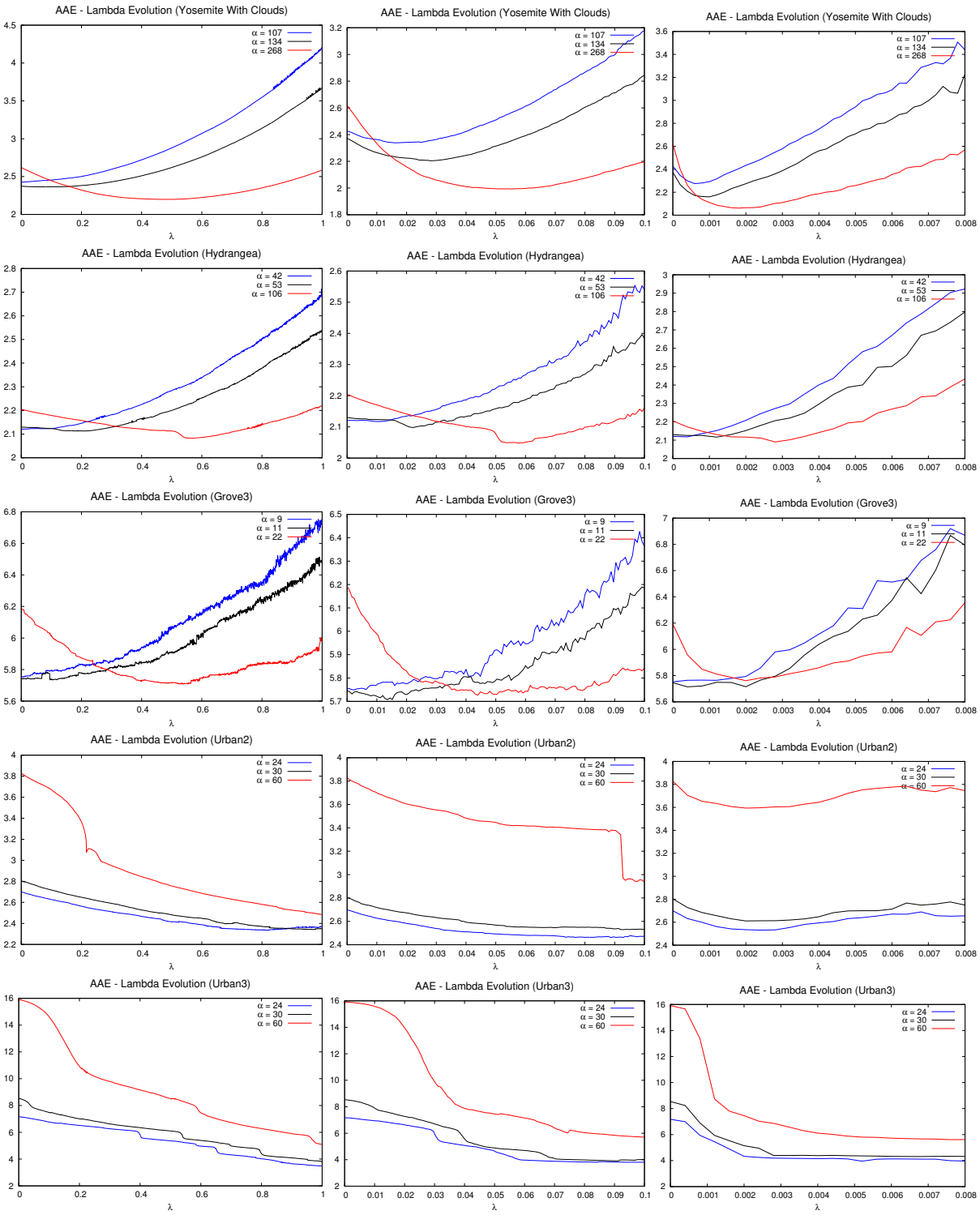


Figure 5.4: Each row depicts the AAE for the sequences using different values of κ: 0.2 in the first column, κ: 1 in the second and κ: 2 in the third.

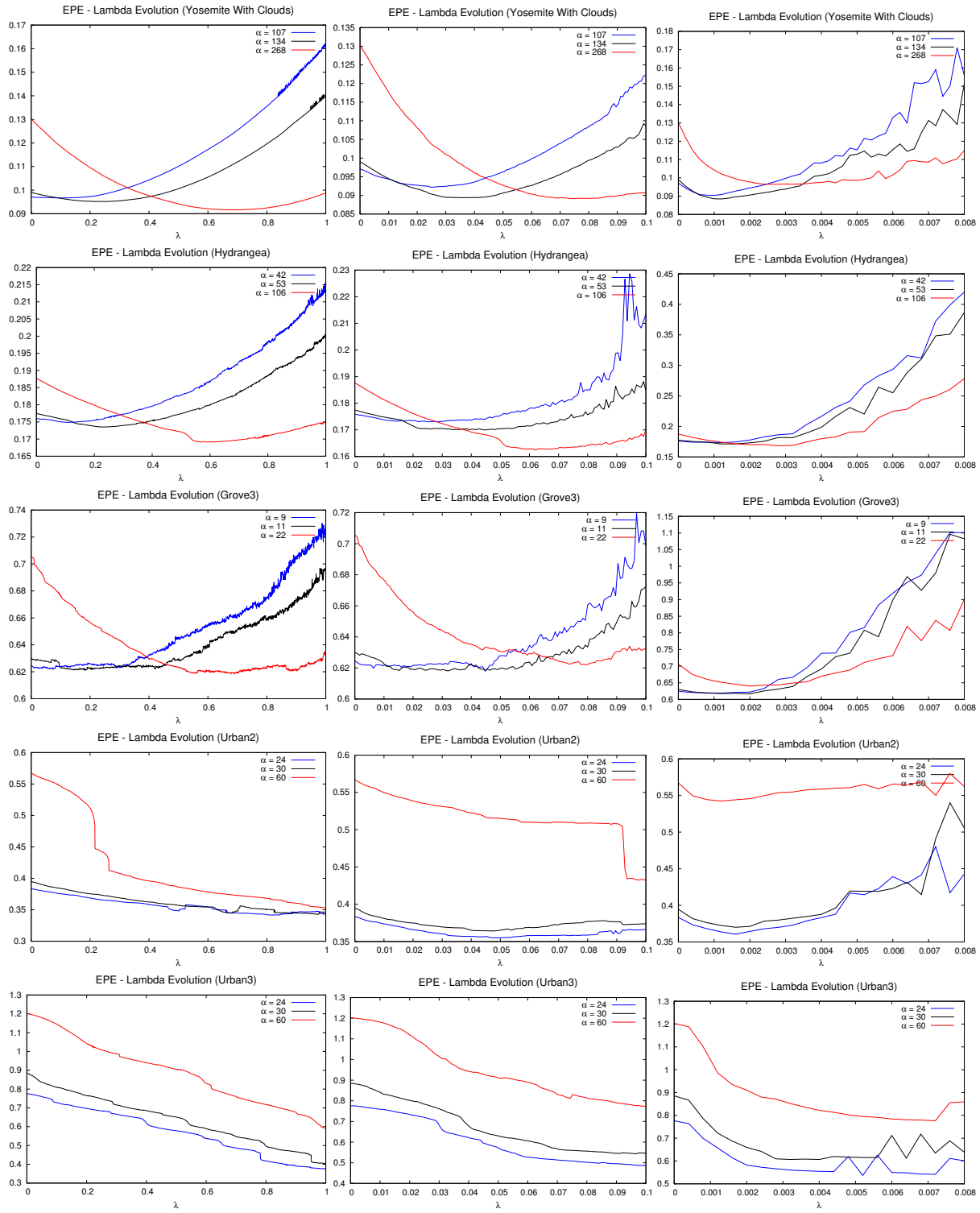


Figure 5.5: Each row depicts the EPE for the sequences using different values of κ: 0.2 in the first column, κ: 1 in the second and κ: 2 in the third.

Nevertheless, we appreciate how the instabilities problem appears when λ is large and numerous outliers appear in the solutions. These are typically located at flow discontinuities and, in the case of the star, also inside the figure, probably as the result of smoothing with the outliers at contours. A similar behavior can be observed in the the other sequences.

In figure 5.8 we show the evolution of the AAE with respect to the discontinuity parameter for the sequences of figures 5.6 and 5.7.

We observe that the exponential clearly improves the basic model of Brox *et al.* represented in the graphics when *DF* uses $\lambda = 0$. The improvement is more important in the geometric figures, where there is an important leap in accuracy after a suitable value. However, we also notice that the solutions rapidly become very unstable for bigger λ values. The best results are normally obtained for large values of α. In the case of Grove2, the best result is obtained for a small α.

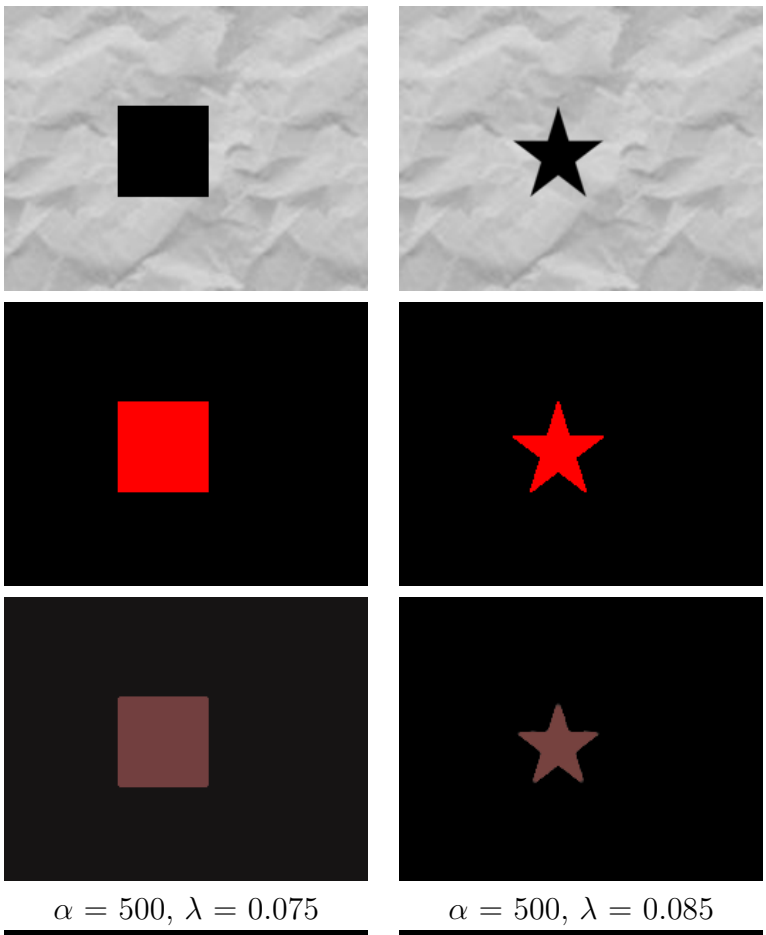
After these experiments, we conclude that the approach is very promising and the decreasing function enhances edge detection. However, it is clear that it strongly depends on the value of the discontinuity parameter. The method works fine in circumstances where the decreasing function does not cancel the regularization term. When this happens, it considerably increases the errors as we have seen in these experiments. This problem is addressed in the *DF-*β and *DF-Auto* alternatives.

5.2.2 Analysis of the *DF-*β Method

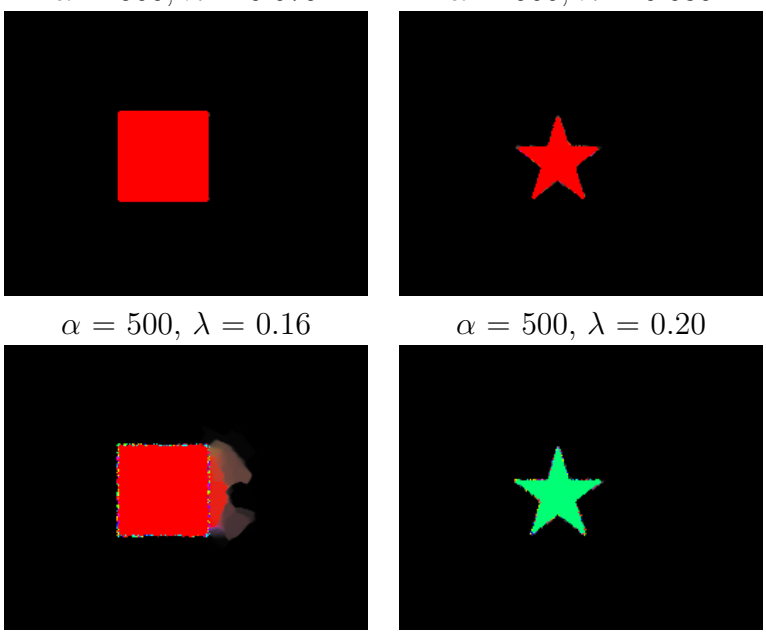
In this section, we analyze the influence of β in the *DF-*β method. We set to zero the gradient parameter (γ) and use large values for α and λ. Then, we increase β from 0.0001 to 0.01 and observe the results. In figure 5.9, we use the synthetic sequence of the double rectangle and two real sequences. In the geometric one, we observe that the enhancement achieved with a small value of β is very good, eliminating the outliers of the *DF* method. However, when $\beta >= 0.001$, the resulting flow is very smoothed without the separation between both rectangles.

Interestingly, we observe in the realistic sequences (mainly in the one with a car) a relative stability in the results for different β values and the blobs of the *DF* solution are clearly removed. We also notice that a bigger parameter produces a more continuous motion field. In fact, the hand of the second sequence disappears when $\beta := 0.01$ and the car contours are dislocated like in Brox solutions.

We may conclude that the value that provides the best results is $\beta := 0.0001$, so we fix it in the experiments.



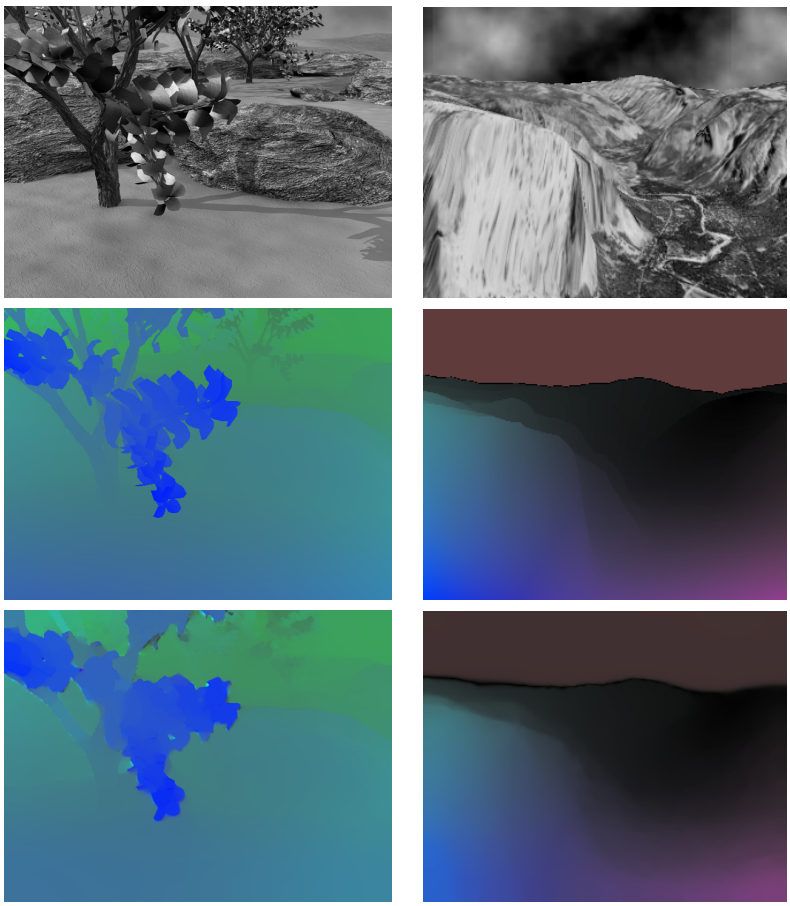
$\alpha = 500,\ \lambda = 0.075$ $\alpha = 500,\ \lambda = 0.085$



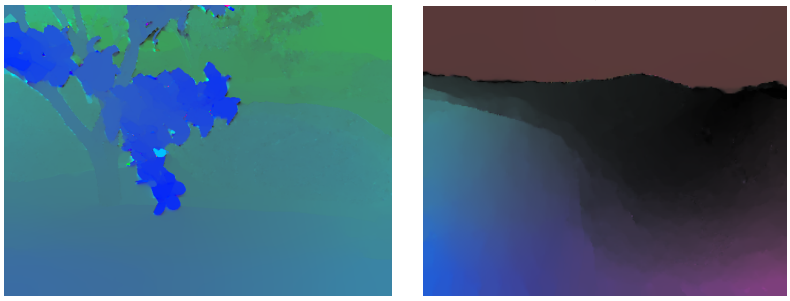
$\alpha = 500,\ \lambda = 0.16$ $\alpha = 500,\ \lambda = 0.20$

$\alpha = 500,\ \lambda = 1$ $\alpha = 500,\ \lambda = 0.54$

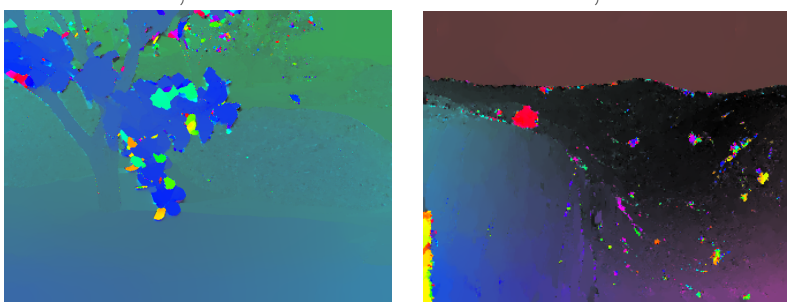
Figure 5.6: Results of the *DF* method for different values of λ. From top to bottom, the first frame of each sequence, the true flow and the optical flows obtained.



$\alpha = 50$, $\lambda = 0.12$ $\alpha = 500$, $\lambda = 0.02$



$\alpha = 50$, $\lambda = 0.34$ $\alpha = 500$, $\lambda = 0.18$



$\alpha = 50$, $\lambda = 0.66$ $\alpha = 500$, $\lambda = 0.8$

Figure 5.7: Results of the *DF* method for different values of λ. From top to bottom, the first frame of each sequence, the true flow and the optical flows obtained.

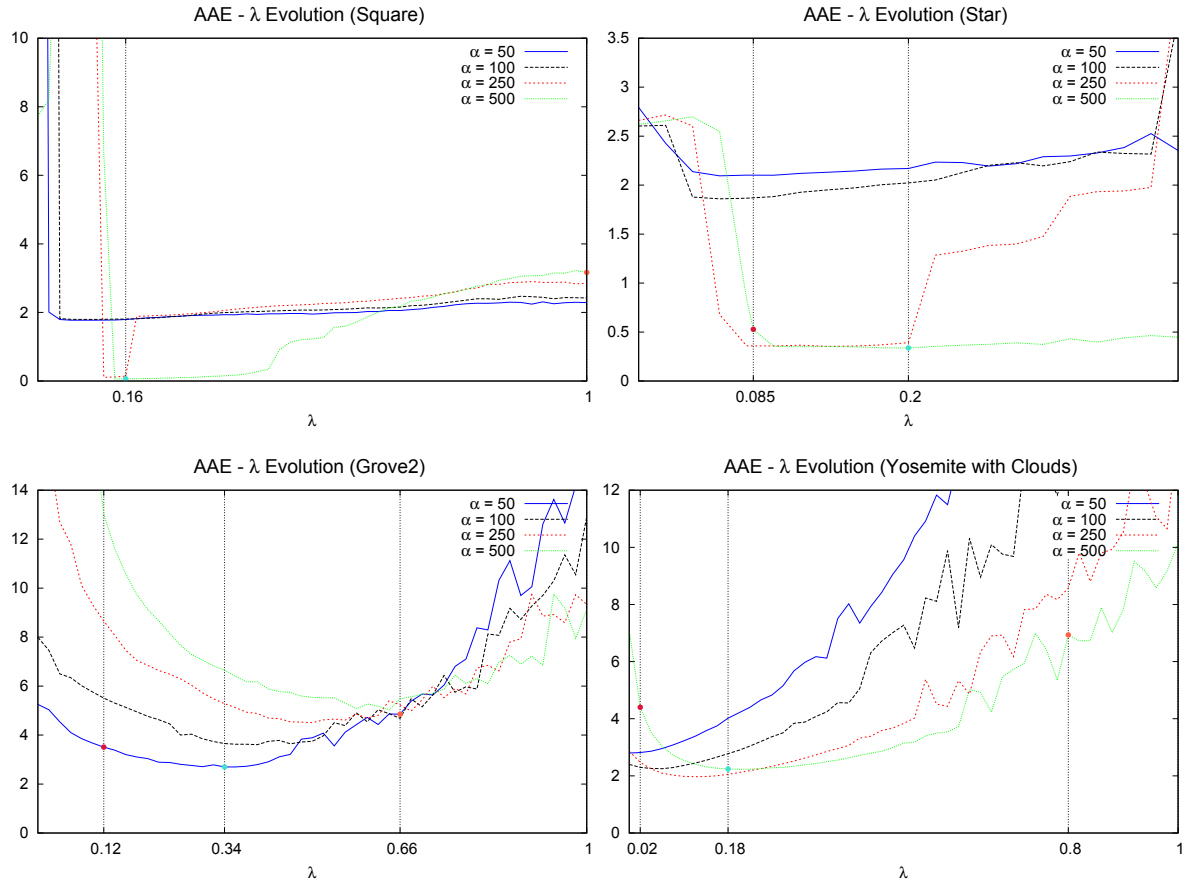


Figure 5.8: Error evolution with respect to λ using the *DF* method. We observe that the parametric range which provides accurate results is small and highly unstable.

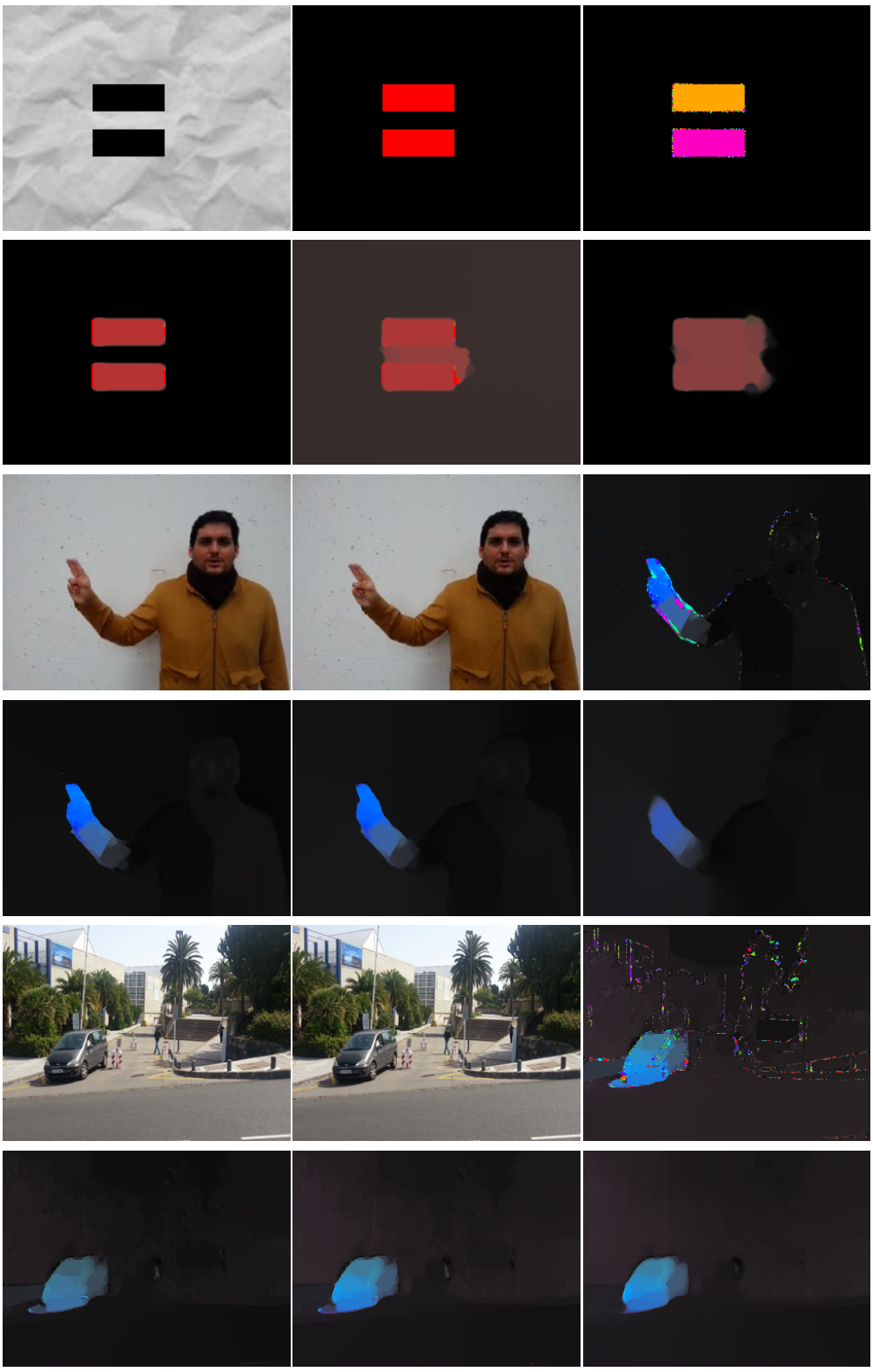


Figure 5.9: Results of the *DF*-β method for different values of λ. The odd rows show the first and second images (or the true flow when it exists) and the optical flow obtained with the basic *DF* method. The even rows show the results for $\beta := 0.0001, 0.001, 0.01$, respectively.

5.2.3 Analysis of the *DF-Auto* Method

The *DF-Auto* approach is, in our opinion, the most interesting proposal due to the automatic setup of λ. In Section 5.1, we observe that the *DF-Auto* uses a ξ parameter to avoid the cancellation of the divergence term. In this section, we analyze the influence of this parameter. We use large values for α and λ in order to study the influence of ξ when outliers appear in the solutions. Figures 5.10, 5.11 and 5.12 show the results for the double rectangle, *Hydrangea* and Bandage 2, respectively.

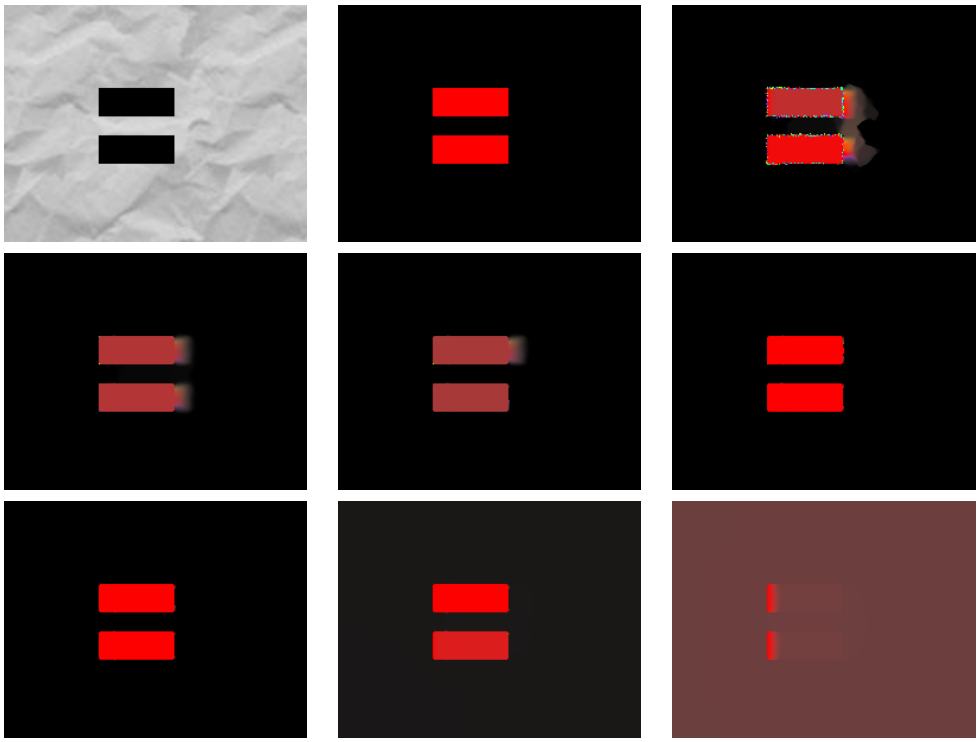


Figure 5.10: Results of the *DF-Auto* method for different values of ξ. First row: the first frame, the true flow and the *DF* solution; Second and third rows: Estimations using *DF-Auto* for $\xi := 0.00001, 0.0001, 0.001$ and $\xi := 0.05, 0.1, 0.99$, respectively.

In figure 5.10, we see that the *DF-Auto* provides better solutions than the original *DF* method. Small values of ξ still yields instabilities in the optical flow, however, when ξ is increased, the outliers progressively disappear. For larger values, the solutions become smoother and, finally, the flow discontinuities are not preserved.

The strategy correctly preserves the flow discontinuities when the corresponding gradients are similar in the whole image. However, when the gradients in the discontinuities are much smaller than the maximum gradient, this method works poorly. In this case, the discontinuities tend to be smoothed due to the difference with respect to the maximum gradient.

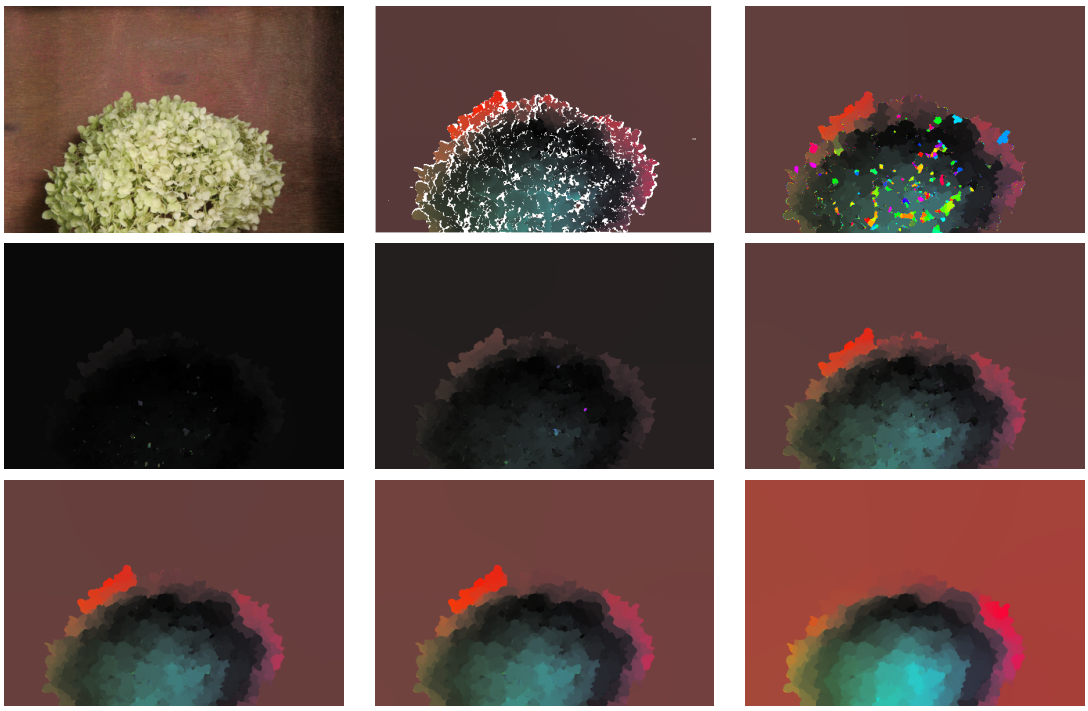


Figure 5.11: Results of the *DF-Auto* method for different values of ξ. First row: the first frame, the true flow and the *DF* solution; Second and third rows: Estimations using *DF-Auto* for $\xi := 0.00001, 0.0001, 0.001$ and $\xi := 0.05, 0.1, 0.99$, respectively.

Figure 5.11 depicts the same evolution but using the *Hydrangea* sequence. The behavior is similar to the previous sequence and the instabilities disappear when we increase the value of ξ. However, we observe a stronger regularization of the motion field. This is due to the big difference between the maximum gradient and the gradients of other discontinuities, that yields the quantization of the flow in a few planar regions. In figure 5.12, the ξ evolution in the bandage 2 sequence is quite stable for the range of $\xi := 0.001, 0.05, 0.1, 0.99$ with a lower degradation than the other sequences.

Once we have finished these experiments, we observe that $\xi := 0.05$ achieves accurate and stable results when using the automatic approach. Therefore, we fix this value for the *DF-Auto*.

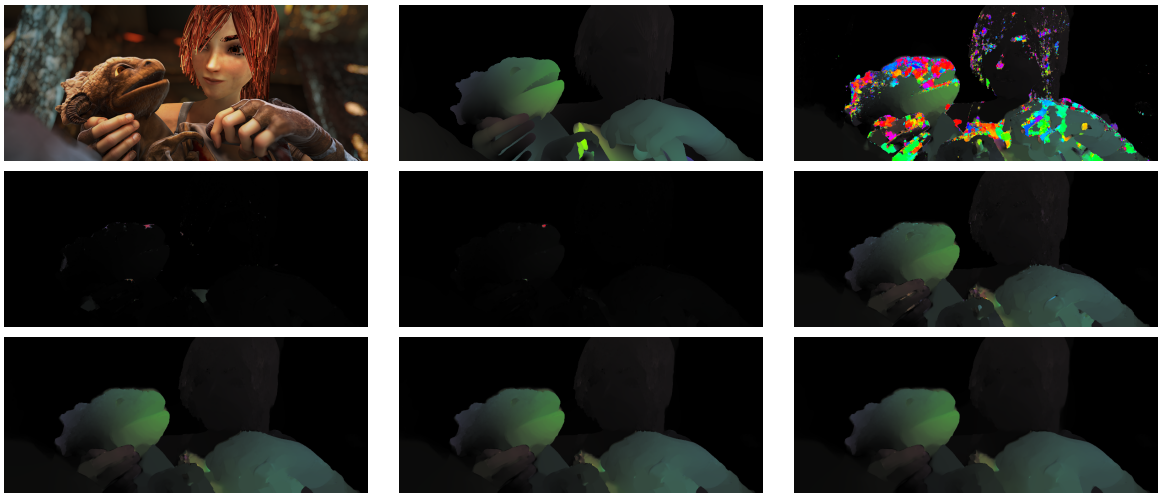


Figure 5.12: Results of the *DF-Auto* method for different values of ξ. First row: the first frame, the true flow and the *DF* solution; Second and third rows: Estimations using *DF-Auto* for $\xi := 0.00001, 0.0001, 0.001$ and $\xi := 0.05, 0.1, 0.99$, respectively.

5.3 Experimental Results

Once analyzed the different approaches, we now compare them and highlight their main features. We include in this analysis the *RADT* method. First, we study the stability of the methods with respect to λ; second, we carry out a comparison between the different diffusion schemes in order to study their dependence on λ and their effects over the regularization. Both studies are focused on the relation between α and the discontinuity parameter. Then, we present numerical results using the best parameter settings for *Middlebury* and *Sintel* datasets.

We finish the comparison with experiments that summarize the features of our proposals respect to Brox and a classical decreasing function, showing their parametric stability and the consequences that they carry out over the flow contours. Then, we analyze the influence of the smoothness weight with respect to different values of γ. The parameters α, γ and λ are modified in each experiment.

5.3.1 Error Evolution with Respect to λ

Here, we compare the EPE stability regarding to the discontinuity parameter for several sequences from the *Middlebury* and the *MPI Sintel* datasets in figures 5.13 and 5.14 respectively.

In the graphics, we depict the error evolution of the *DF* method with a blue line. According to the results, the exponential function consistently improves the Brox method (*DF* with $\lambda = 0$), specially in the graphics of *Shaman 2*, *Bamboo 1*, *Urban3* or *RubberWhale*. Nevertheless, once we achieve the best solution, the graphics are quite unstable and the blobs begin to appear.

We notice that the results of the *Shaman 2* sequence are better, providing more stable results for a large range of λ values. In this case, its images present very bright and dark areas allowing to increase the parameter and still yielding good solutions.

In contrast, we observe a smoother evolution (violet line) when we include the small constant of the *DF-β* approach. In this case, the effects over the motion field are quite similar even with an extreme parameters. The regularization never gets canceled due to the β constant and, for most of the sequences, the results improve when *DF* strongly deteriorates the solutions.

Interestingly, the results of the *DF-Auto* proposal (red line) are close to the best solutions that an exponential scheme can provide. The diffusion is controlled locally for every pixel using the histogram information. It has the positive effect of adapting the discontinuity parameter to homogeneous and discontinuous regions.

The automatic method adjusts its values according to the pixel information. As a consequence, the decreasing function is more incisive in areas with strong gradients while

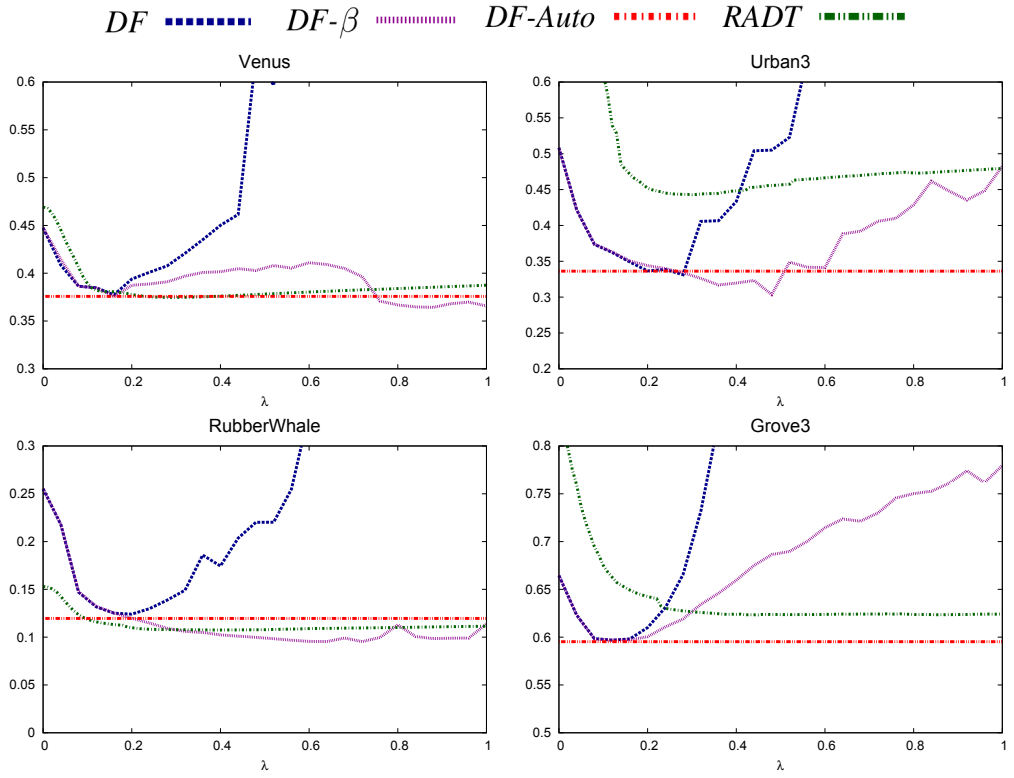


Figure 5.13: Average End Point Error (EPE) evolution with respect to λ for *MPI Sintel* and *Middlebury* sequences. The use of decreasing functions improves the solutions obtained with the Brox approach (*DF* with $\lambda = 0$).

it is more permissive in zones with low variations. This behavior is not possible for the *DF* and *DF-*β methods. These approaches use a unique parameter value for the whole image. Thus, the control of the diffusion only depends on the gradient strength.

Finally, a green line depicts the evolution of the *RADT* method. In this case, the regularization becomes quite similar to the Horn-Schunck method when $\lambda = 0$ and makes that the starting error is usually bigger than other strategies. For this reason, we have used a range of $\lambda \in [0, 50]$ and we have normalized the results between $[0, 1]$ to fit into the graphics. However, the *RADT* approach provides the most stable evolution with errors that remains almost constant for large values of λ. This is reasonable, because, unlike the other schemes, there always exists a regularization along the isocontours of the objects. For a sufficiently large value of λ, the robust function cancels the regularization across the borders of the object. Then, the results only depend on the isocontour smoothing, which is always the same regardless of λ.

From these graphics, we conclude that the basic *DF* scheme is very unstable and the range of interest values is very small in practice. In contrast, the *DF-*β and

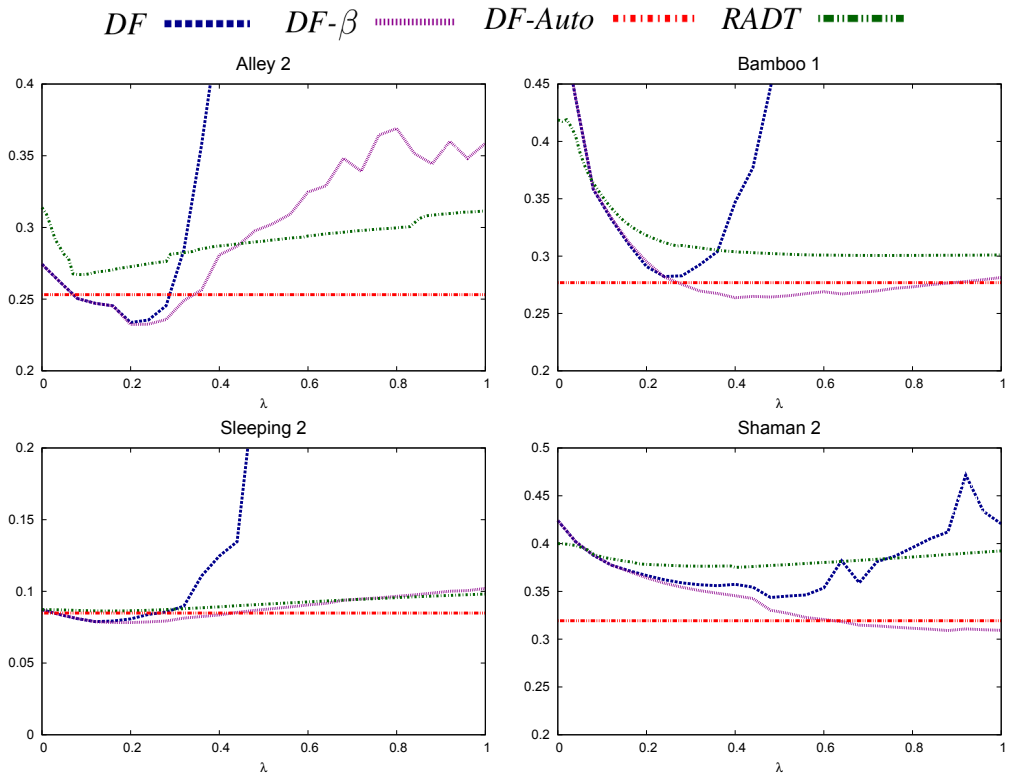


Figure 5.14: Average End Point Error (EPE) evolution with respect to λ for *MPI Sintel* sequences. The use of decreasing functions improves the solutions obtained with the Brox approach (*DF* with $\lambda = 0$).

RADT approaches are much more stable and reduces the parameter setup. The *DF-β* preserves the accuracy of the basic method but yielding a better stability. Even, in some sequences the error improves. The *DF-Auto* method is close to the best result for many of the sequences and, in most cases, it provides results which are near the best solutions. On the other hand, the *RADT* technique is very stable in general. However, the accuracy is typically worse. The use of more advanced techniques, like in [Zimmer11], may turn this method more competitive.

5.3.2 Comparison of the Regularization Strategies

Next, we compare the effects over the motion fields for several sequences using the different strategies and their EPE graphics with respect to λ. The first row of figures 5.15, 5.16, 5.17 and 5.18 show the original image and the resulting graphics for the *rectangles*, *Yosemite with clouds*, *Grove2* and *Alley 1* sequences. The second and third rows show the different flow fields and the true flow.

Note that, in these flows, we have intentionally taken a value of λ which is slightly bigger than its optimal value for comparing the stability of *DF*, *DF-*β and *RADT*. These flows are represented in the graphics by the vertical lines and the X marks. Note that these points do not always appear for the *DF* graphic because the error is normally too big.

In general, the flow discontinuities obtained with the Brox method are not aligned with the object contours. For instance, the method cannot distinguish between the two rectangles of figure 5.15 and the leafs in *Grove2* and the hair tufts of the woman in *Alley 1* are heavily smoothed. The motion contours are better preserved using the other strategies, principally dealing with the geometric figures.

On the other hand, these four figures are a good example of the high parametric dependence of the *DF* method, where numerous instabilities appear while the others schemes offer a reasonable quality. In fact, the solution of our first proposal is quite similar to the *DF* method, but with an important reduction of blobs.

The *RADT* strategy provides good solutions, like for instance, in figure. 5.18 where the method correctly detects the woman's arm and the apple in her hand. Nevertheless, the diffusion is not completely stopped at the object contours, especially at the skyline in the *Yosemite* sequence. Note that this behavior is similar in the *Shaman 2* results showed in figure 5.3, where the small gradients do not allow to steer the diffusion process conveniently.

Finally, we observe an interesting issue in the graphic of figure 5.17 that we must consider. According to the error evolution, the differences between the Brox results and the other approaches are not remarkable. However, the flow fields demonstrate better boundaries definition but, with some small instabilities that worsen the average error of the whole flow. In this sense, it is possible for an oversmoothed solution to have the same error as a flow which is overall more accurate but with some outliers due to instabilities.

In summary, we observe the same behavior as in the previous section. From the *Rectangles* sequence we can appreciate the limitations of the Brox method and the benefits of using the other strategies. The motion contours are much better preserved even though the EPE errors are not so different with respect to Brox. The graphics show the instability problems of the *DF* method and we can clearly see the blobs in the flow images. This means that there is a correlation between the choice of λ, the appearance of instabilities and the increase of EPE errors. The *RADT* method preserves good motion discontinuities but the flow field is smoother than *DF-*β and *DF-Auto*. However, the latter still introduce some instabilities around the contours, probably at occluded regions, which are not present at *RADT*.

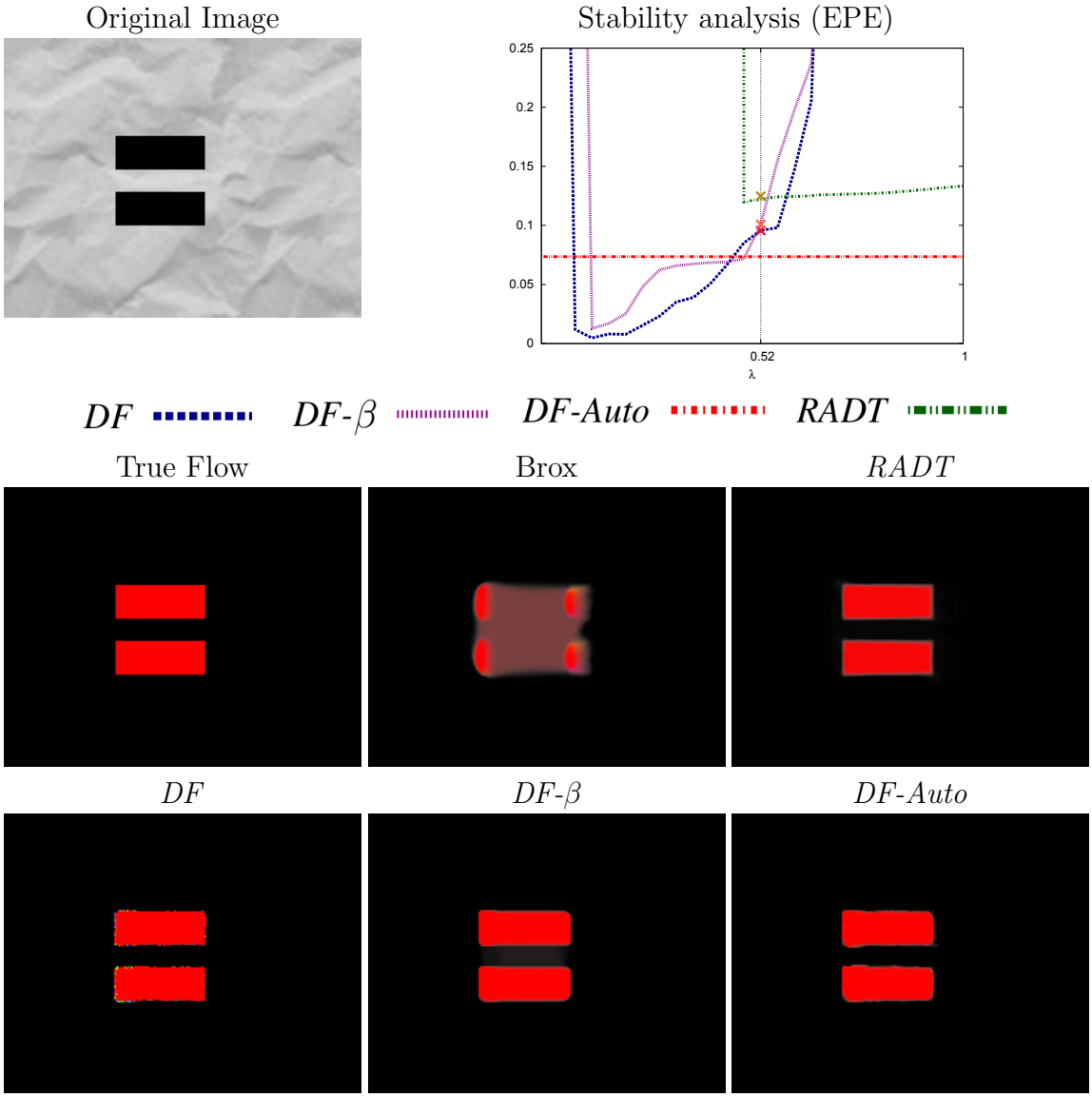


Figure 5.15: First row: The *Rectangles* sequence and its EPE graphic with respect to λ. Second row: the true flow and the solutions obtained with Brox and *RADT* methods, respectively. Third row: *DF*, *DF-*β and *DF-Auto* solutions using the λ value that appears in the graphic (0.52). The *DF* detects the discontinuities but introduces some instabilities. The *DF-*β and *DF-Auto* variants eliminate these instabilities but create slightly rounded flows. The *RADT* approach creates straight edges but these are not so sharp.

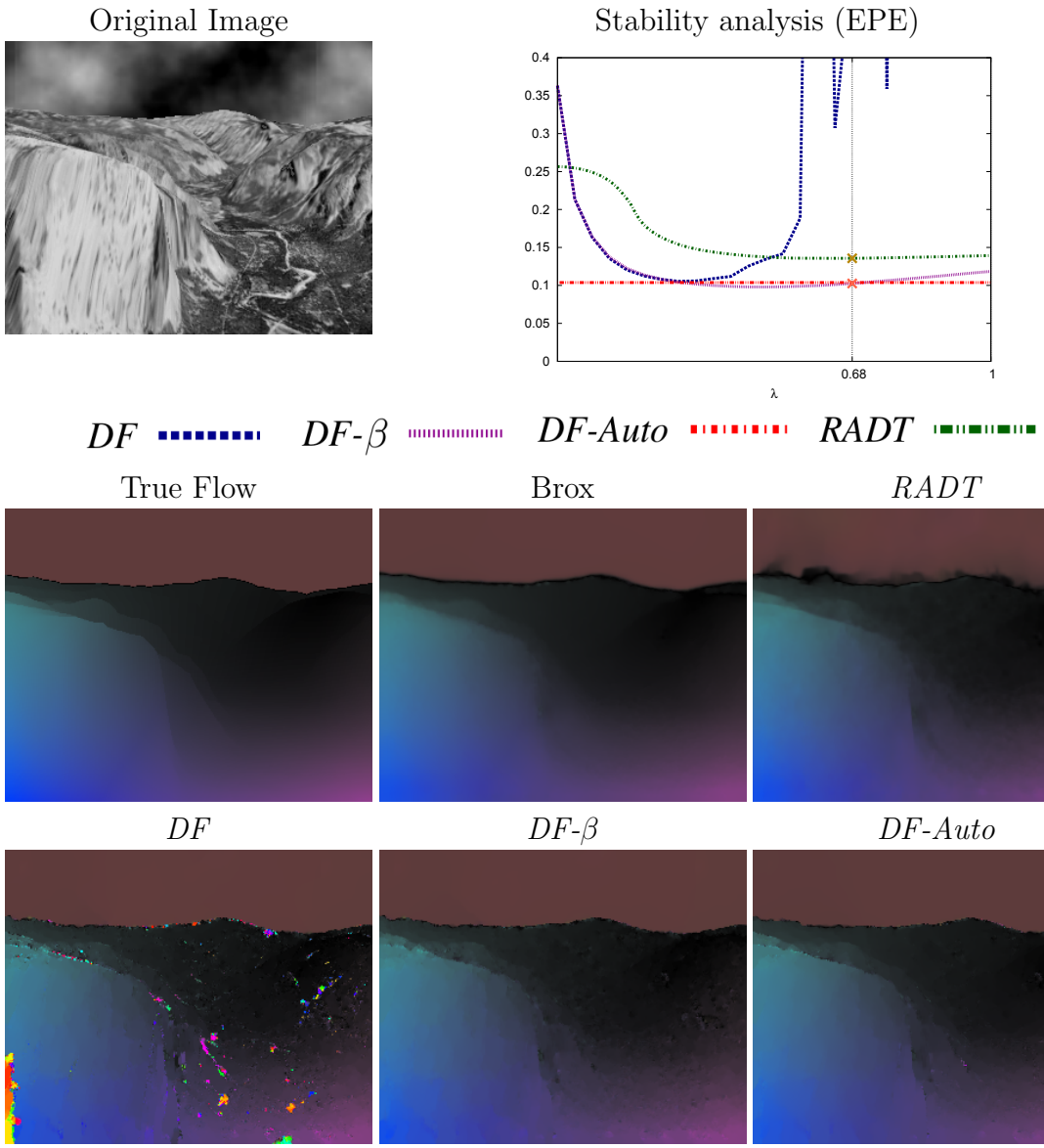


Figure 5.16: First row: *Yosemite* sequence and its EPE graphic with respect to λ. Second row: the true flow and the solutions obtained with Brox and *RADT* methods, respectively. Third row: *DF*, *DF-*β and *DF-Auto* solutions using the λ value that appears in the graphic (0.68). The flow field obtained by Brox visually resembles the true flow, except in the skyline and the mountain discontinuities. The textures in the mountains create slightly segmented flows in the other methods. *DF* also introduces instabilities in high gradient regions. *DF-*β and *DF-Auto* eliminate these instabilities. *RADT* provides a good flow except in the skyline.

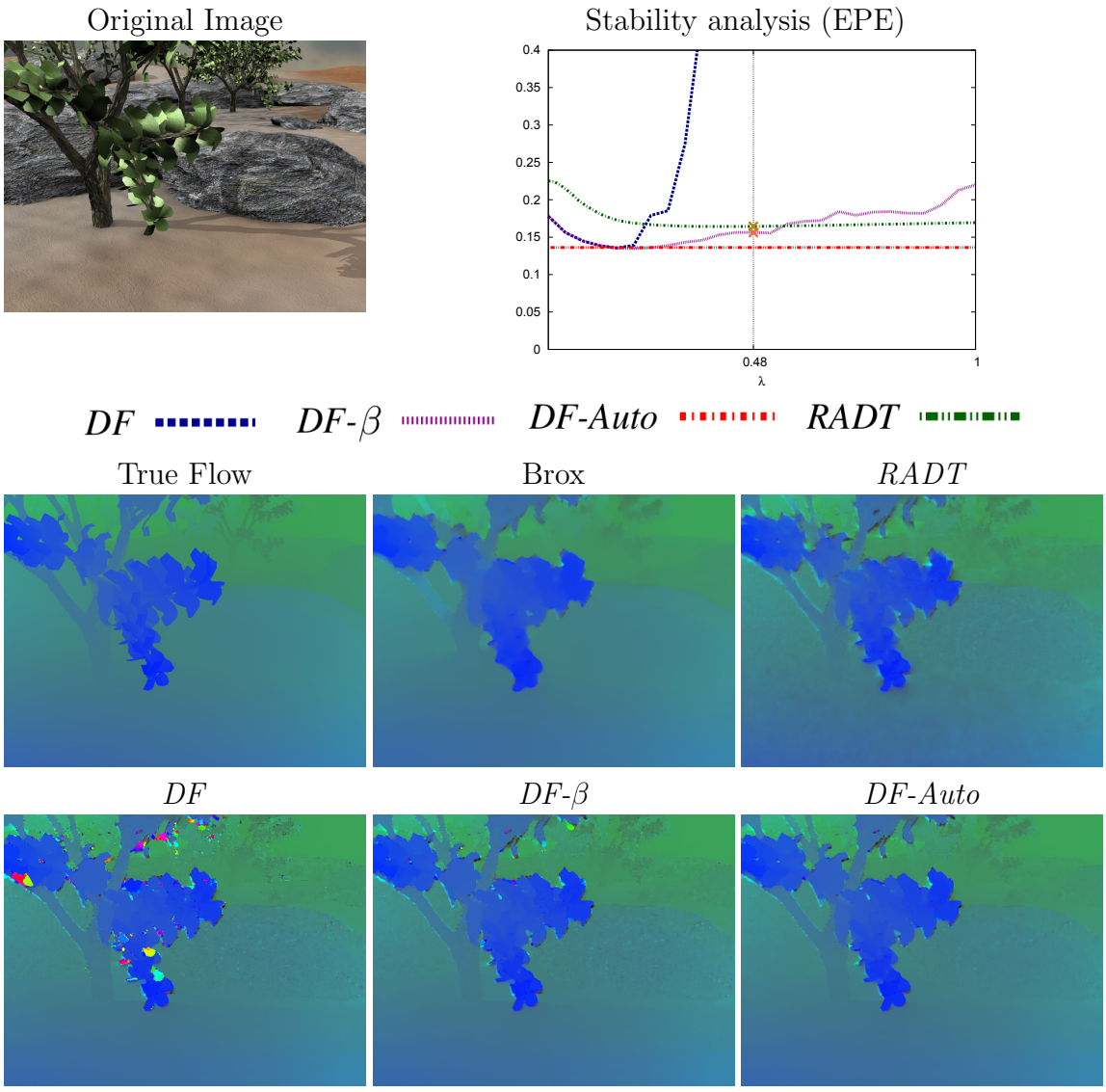


Figure 5.17: First row: *Grove2* sequence and its EPE graphic with respect to λ. Second row: the true flow and the solutions obtained with Brox and *RADT* methods, respectively. Third row: *DF*, *DF-*β and *DF-Auto* solutions using the λ value that appears in the graphic (0.48).In contrast to the Brox method, the discontinuity-preserving strategies offer an accurate flow at motion edges. In general, the behavior is similar to *Yosemite* in Fig. 5.16.

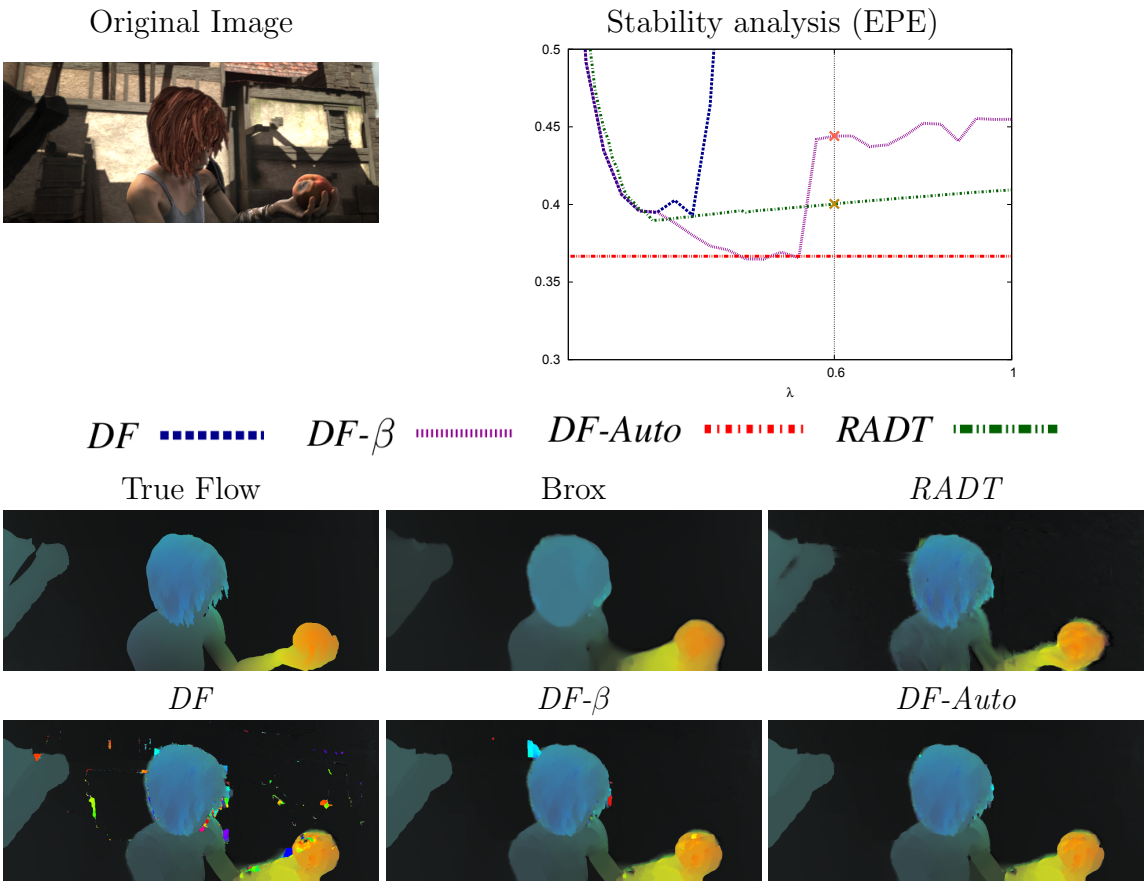


Figure 5.18: First row: *Alley 1* sequence and its EPE graphic with respect to λ. Second row: the true flow and the solutions obtained with Brox and *RADT* methods, respectively. Third row: *DF*, *DF-*β and *DF-Auto* solutions using the λ value that appears in the graphic (0.6). The main differences are located in the woman's arm and in the apple.

5.3.3 The Improvement in the Parametric Stability

Next, we compare our two proposals with respect to the original model (Brox) and the pure decreasing function of the *DF* method. First, we use the synthetic sequences of *Urban2* and *Shaman 2* for showing the behavior of the methods with respect to the discontinuity parameter in figures 5.19 and 5.20, respectively. The first row presents the original image and the average End-Point Error (EPE) evolution. The second row depicts the true flow, the result of using Brox and the flow obtained with the *DF-Auto* approach. The third and fourth rows show the motion fields for *DF* and *DF-*β strategies when increasing λ.

In figures 5.21 and 5.22, we reproduce a similar experiment but using a pair of frames from the sequences of *Rheinhafen* and *Ettlinger-Tor*. In all these figures, our intention is to show the parametric stability and the effects over the flow contours for $\lambda := 0.1, 0.5, 1.0$.

Finally, we analyze the influence of the smoothness weight with respect to different values of γ.

Once again, the Brox solutions have problems dealing with motion contours, except in the *Urban2* sequence where its flow edges are better aligned with the object borders. We also notice that a strong regularization has made disappear some cars in the *Rheinhafen* and *Ettlinger-Tor*.

Interestingly, our exponential strategies are accurate at the flow edges. The corners of the buildings in *Urban2* are better preserved compared with the Brox solution and the borders of the vehicles in *Rheinhafen* and *Ettlinger-Tor* sequences are better defined. However, we also observe a *staircasing* problem in the floor of the street in *Urban2*. This negative effect occurs because the decreasing function depends on the image gradient treating some image borders as flow edges.

On the other hand, the graphics reaffirm that the *DF* method is highly unstable once the best flow is reached while our proposals present a good stability. In fact, the *DF-*β approach almost eliminates the instabilities or, at least, they are strongly reduced as in the *Urban2* sequence.

The automatic parameter of the *DF-Auto* method adapts its value according to the image gradient providing good optical flows without relevant blobs. This simplifies the parametric configuration required by the other alternatives.

Regarding to the parametric configuration, our previous experiments where centered in the relation between the regularization parameter and λ. Now, the purpose of the figures 5.23 and 5.24 is to observe the dependency of the exponential strategies with respect to the gradient parameter.

We use the synthetic sequences of *Grove2* and *Urban3* from *Middlebury* and *Alley 1* and *Shaman 2* from the *Sintel* dataset. The *DF*, *DF-*β and *DF-Auto* methods are represented with a blue, red and green line, respectively.

The first and second columns show the EPE evolution with respect to α for $\gamma = 0$ and $\gamma = 1$. In chapter 4, we concluded that $\gamma = 7$ provides the best results for the entire *Middlebury* dataset. We use this value for the graphics of the third column of both figures. We fix $\lambda := 0.2$ for *DF* and *DF-*β.

According to the results, the behavior of *DF* and *DF-*β are similar in many cases. The results of *DF-Auto* are more stable in general, especially in the *Urban3* and *Shaman 2* sequences. This means that the search space of the α and γ parameters is reduced for the *DF-Auto* method, so these are easier to configure.

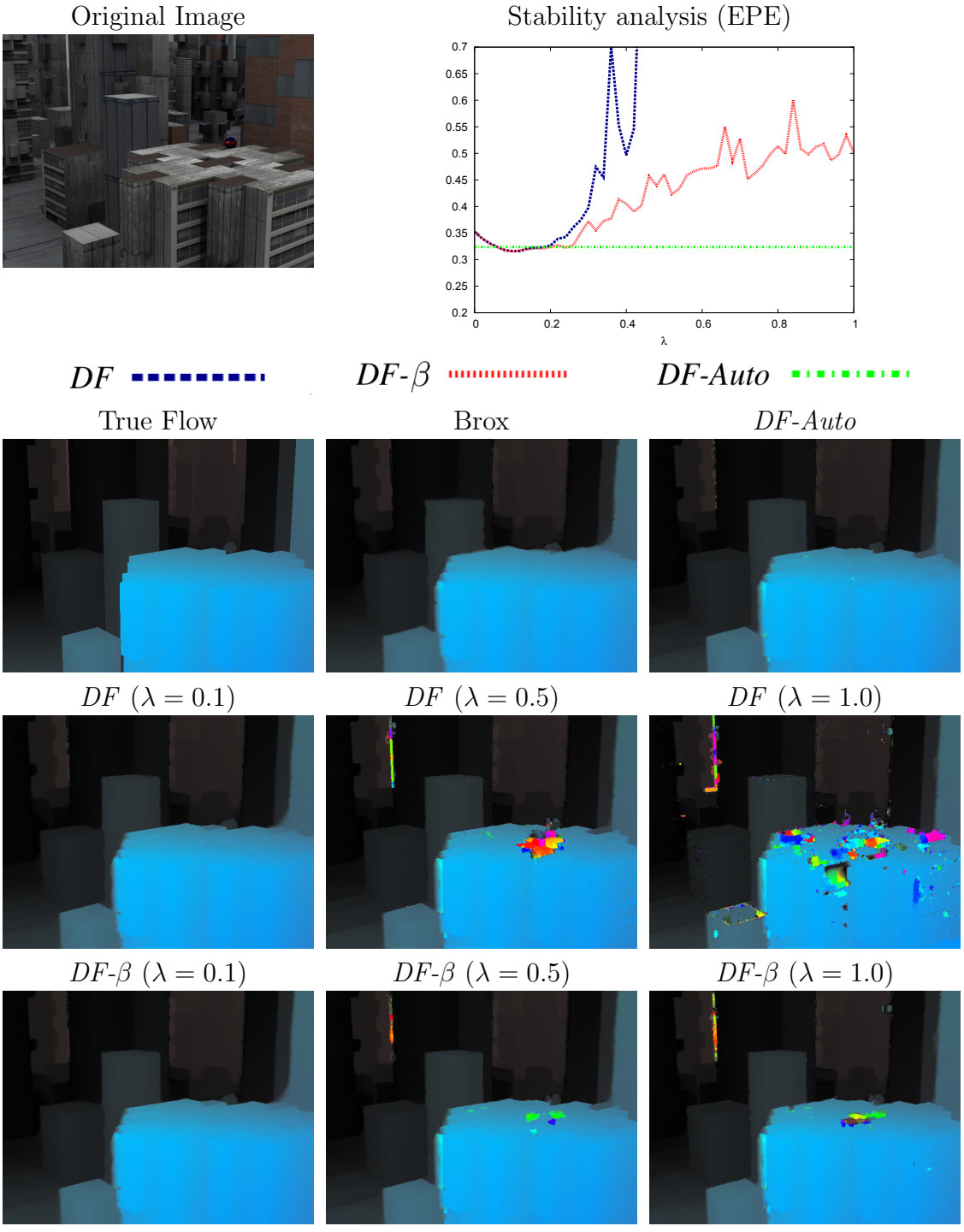


Figure 5.19: *Urban2* sequence. First row: original image and its EPE graphic with respect to λ. Second row: the true flow and the solutions obtained with Brox and *DF-Auto* methods, respectively. Third and fourth rows: Flow fields for increasing values of the λ parameter (*DF* and *DF-*β approaches, respectively). We observe fewer artifacts over the flow with the new proposals. The automatic approach finds a well-preserved solution.

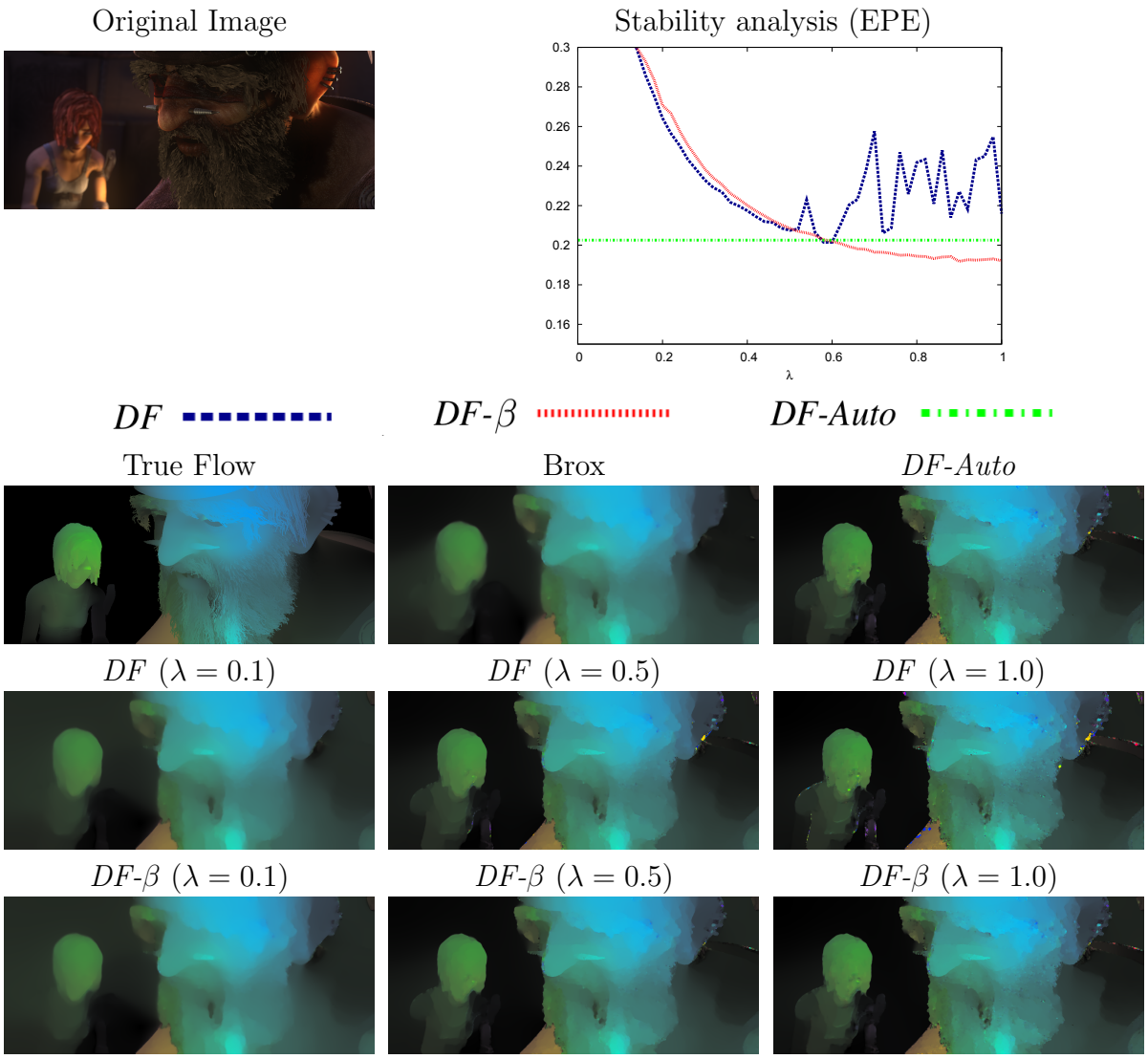


Figure 5.20: *Shaman 2* sequence. First row: original image and its EPE graphic with respect to λ. Second row: the true flow and the solutions obtained with Brox and *DF-Auto* methods, respectively. Third and fourth rows: Flow fields for increasing values of the λ parameter (*DF* and *DF-*β approaches, respectively). We can observe that *DF-Auto* adapts the discontinuity parameter to achieve a good preservation of the motion contours.

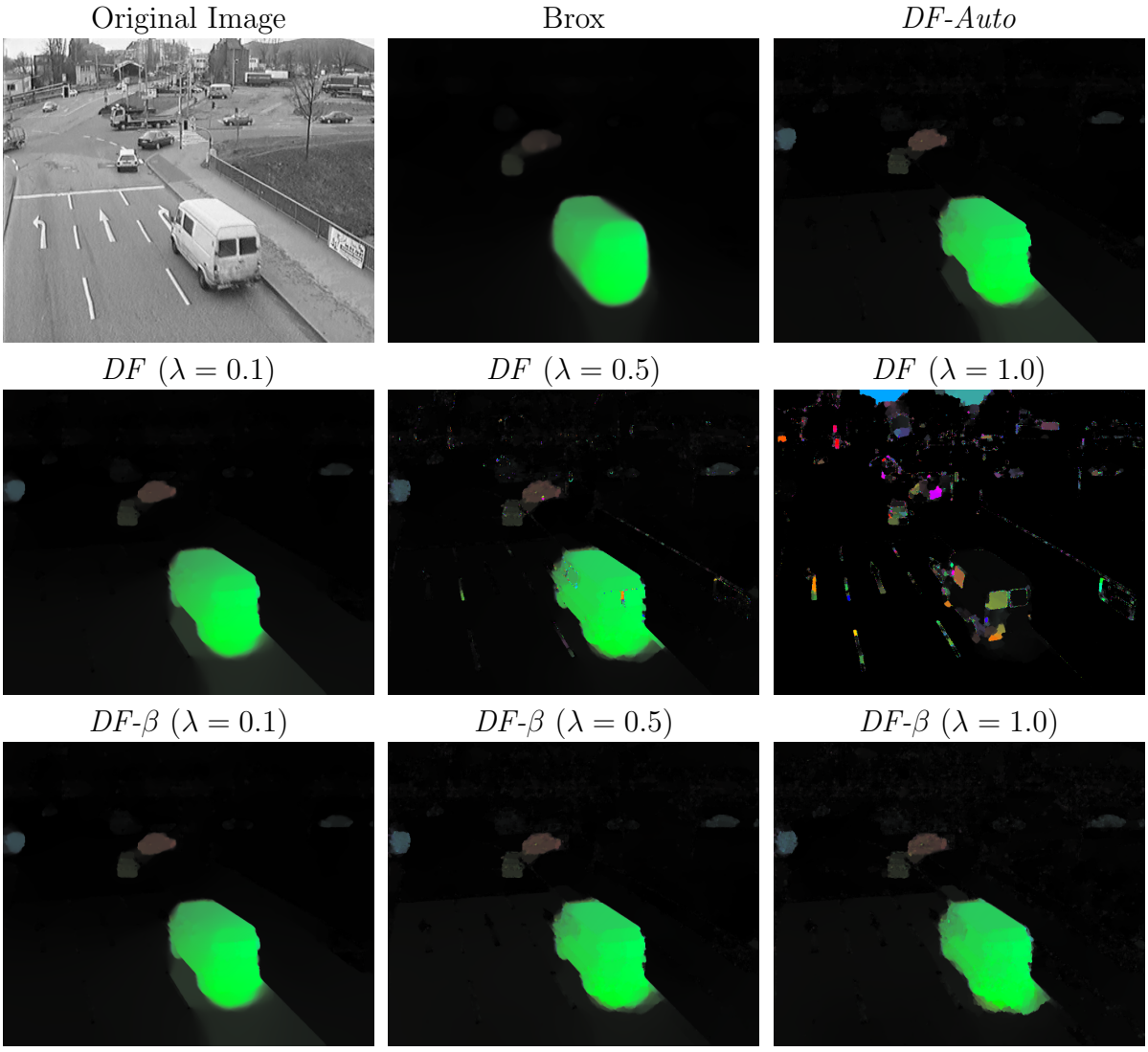


Figure 5.21: *Rheinhafen* sequence. First row: Original image, Brox and *DF-Auto* solutions. Second and third row: Flow fields for increasing values of the λ parameter (*DF* and *DF-β* approaches, respectively). We observe an interesting stability of the λ parameter when using *DF-β* approach. The new proposals present a better definition of the motion contours in comparison of Brox method.

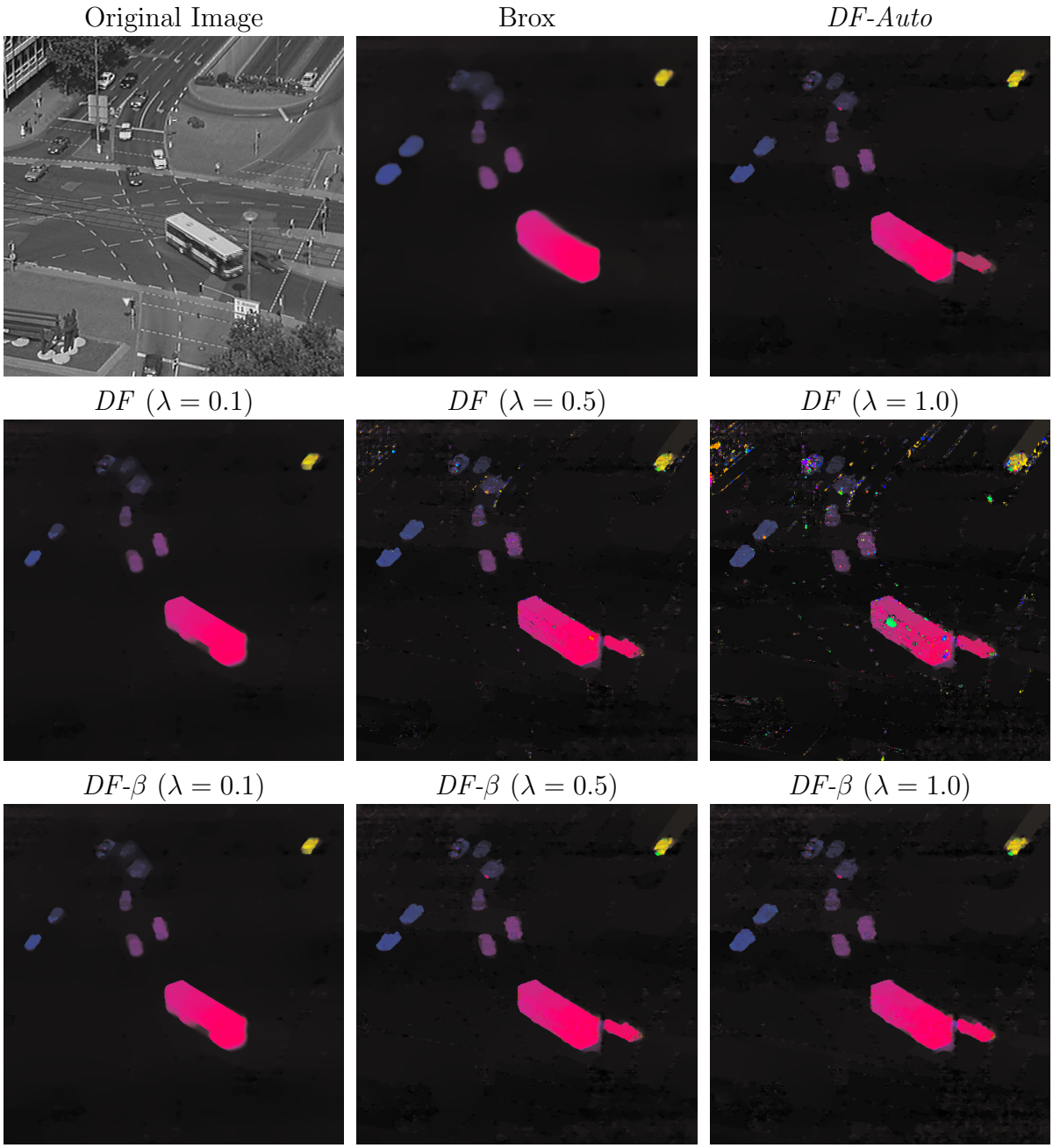


Figure 5.22: *Ettlinger-Tor* sequence. First row: Original image, Brox and *DF-Auto* solutions. Second and third row: Flow fields for increasing values of the λ parameter (*DF* and *DF-*β approaches, respectively). Some cars disappear in the flow because of a strong regularization when using Brox. However, the exponential methods preserve these vehicles and allow an improvement in the object borders.

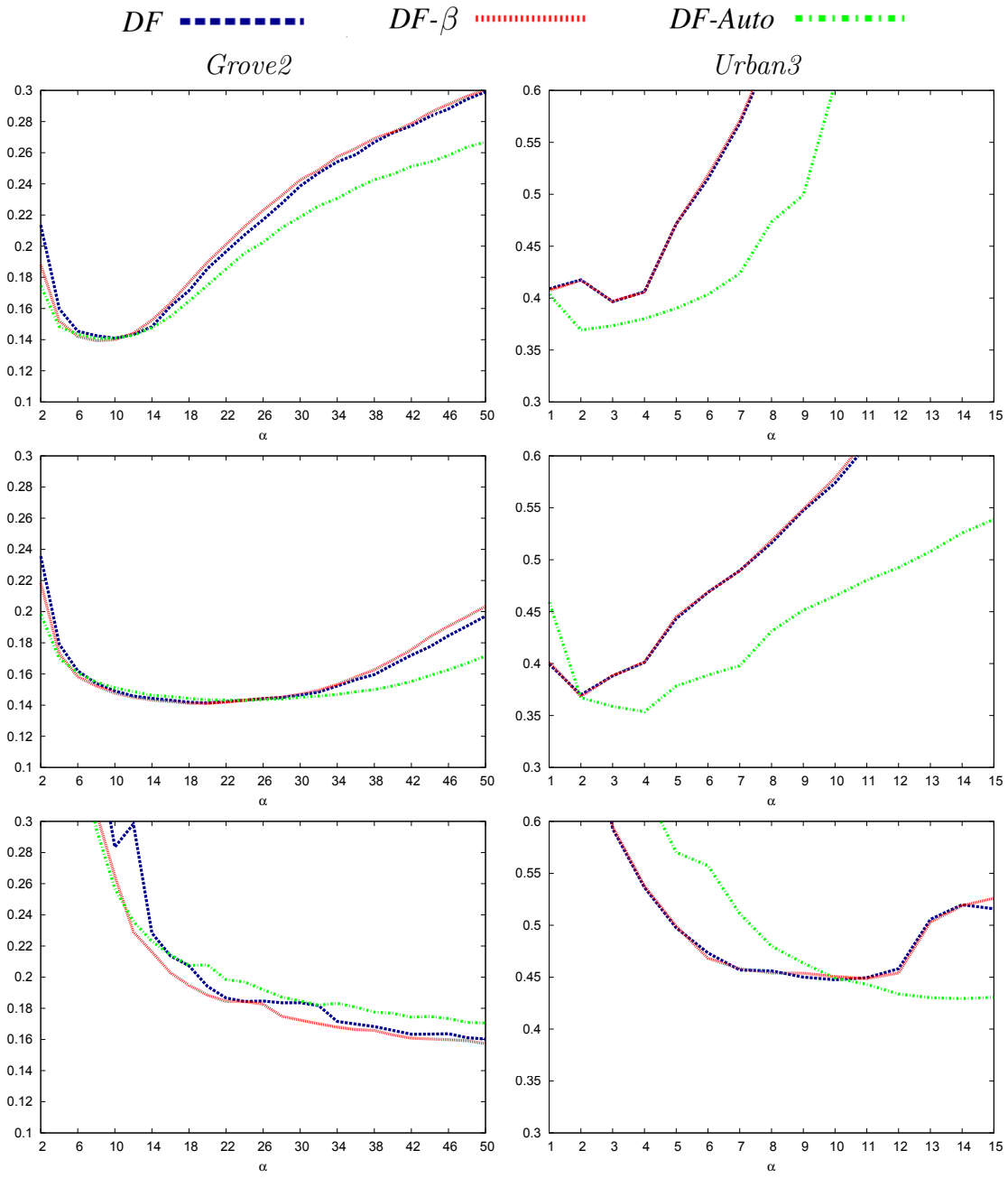


Figure 5.23: EPE evolution with respect to α for some *Middlebury* sequences. From top to bottom, the γ parameter is 0, 1 and 7. We settle $\lambda = 0.2$ for *DF* and *DF-β* methods while it is automatically calculated in *DF-Auto*.

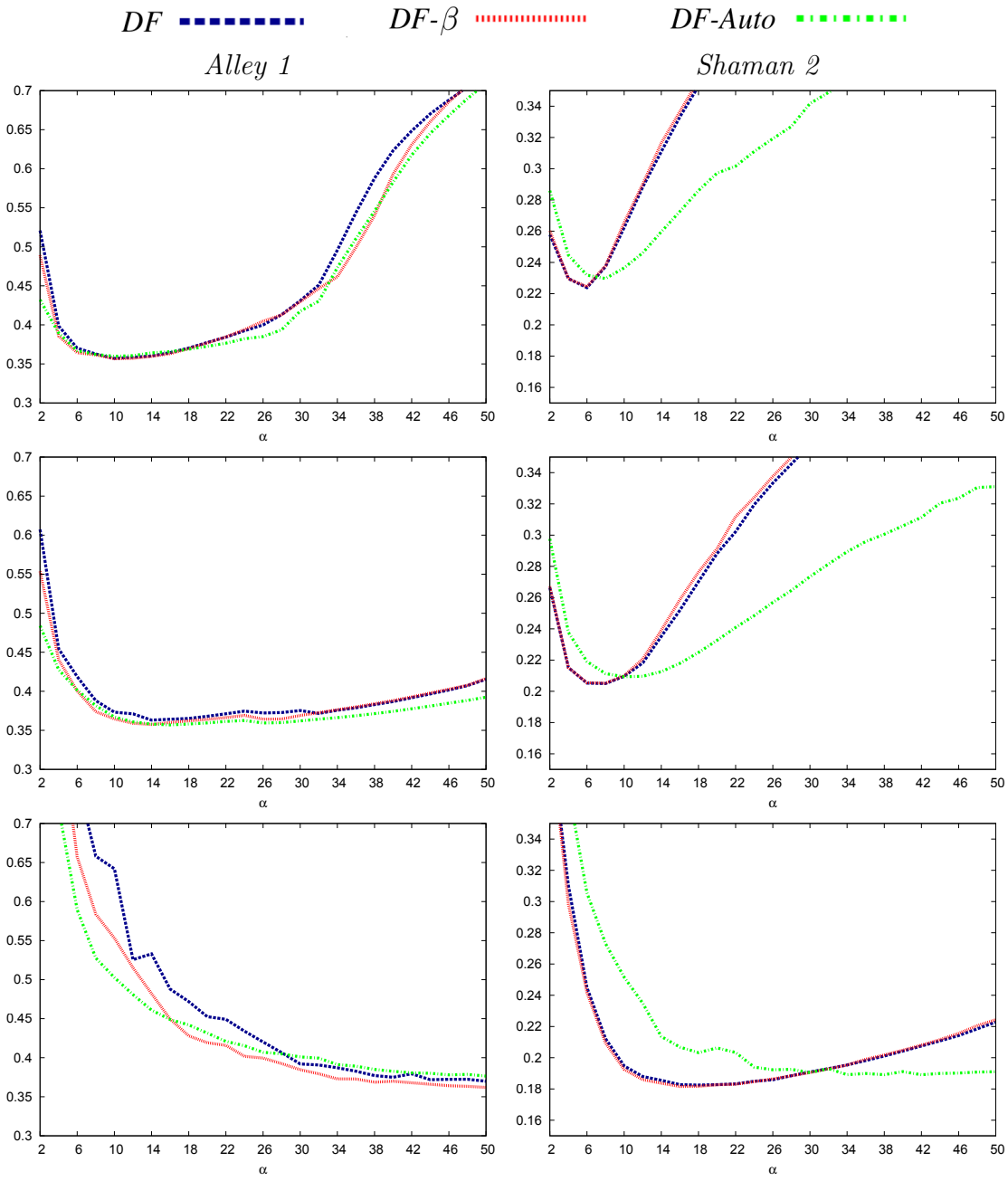


Figure 5.24: EPE evolution with respect to α for some *Middlebury* sequences. From top to bottom, the γ parameter is 0, 1 and 7. We settle $\lambda = 0.2$ for *DF* and *DF-*β methods while it is automatically calculated in *DF-Auto*.

5.3.4 Regularization Strategies for Stereomatching

In the previous experiments, we used standard sequences widely seen in optical flow studies. Next, our intention is to show examples of the regularization strategies dealing with a stereo dataset. In this case, the problem differs because now, we have two views of a static scene taken from different viewpoints.

Figures 5.25, 5.26, 5.27, 5.28 and 5.29 present a similar structure with respect to the previous figures but using stereo pairs from the CMLA dataset [Dagobert] and a new color scheme for the vectorial representation. In particular, we calculate optical flows with the shiny versions of *Salon*, *Pillar*, *Oranges*, *Bastet* and *Shrub* sequences where the camera makes a fronto-parallel displacement of 50 pixels. In the experiments, we set $\alpha := 25$, $\gamma := 1$ and $\lambda := 0.1, 0.5, 1.0$ for the non-automatic approaches. In Brox, we settle $\alpha := 18$ and $\gamma := 7$ according to the conclusions of chapter 4. These experiments have been made using the IPOL demos of Sánchez *et al.*[1] [Sánchez13b] and Monzón *et al.*[2] [Monzón16b]. In this sense, we renamed the Brox results as IPOL-Brox.

As expected, the *DF* approach is very unstable where the new proposals are not. The automatic strategy achieves very good results at the boundaries and preserves many details despite of *staircasing* effects in some areas. On the other hand, *DF-*β is very stable and the risk of failures due to the discontinuity parameter is not too elevated.

For instance, in figure 5.25, the objects have been detected with precision. We note that, even small artifacts such as the sticks of the center table and the lamps, present a good definition of their contours. A similar effectiveness appears in the sequence of the pillar (figure 5.26). In fact, these two sequences are very similar.

In constrast, IPOL-Brox does not have the *staircasing* problem. In both figures, its TV scheme offers an interesting accuracy and the rounded shapes are not too pronounced like in other sequences. However, the accuracy of the automatic approach is better in the arm-chairs and the lamps.

In figure 5.27, the limits of the oranges are well defined and the occlusions has a limited impact in the flow fields. The regularization parameter allowed to fill the information inside these regions with the correct values where Brox introduces some mistakes and does not correctly separate the oranges. Here, although the increment of λ for the *DF* strategy derived into instabilities at the contours, it is also true that the method is much stable regarding to other occasions.

On the other hand, despite of some small incongruences in the left arm of the Bastet depicted in figure 5.28, many details such as the diadem, the nose and the clothes of the statue are well preserved with *DF-Auto*. Even, we can observe the bracelets of the right arm. These details are not clear in the flow obtained by the IPOL-Brox method.

[1] http://www.ipol.im/pub/art/2013/21/
[2] http://www.ipol.im/pub/art/2016/172/

The *DF-β* and *DF-Auto* solutions in figure 5.29 have a few errors in the leafs. Fortunately, these instabilities does not appear too much and, in contrast, the small details of the arbust are usually well preserved. We also observe a *staircasing* problem in the floor. In this zone, IPOL-Brox presents a better behavior. In these last two figures, it is clear that the *DF* method only works well enough for small values of λ. If we overestimate the parameter, the motion fields are very poors.

Finally, we make a brief comparison between different optical flow methods using their corresponding IPOL demos. Figure 5.30 shows the results for the Horn and Schunck method using the demo[3] of Meinhardt-Llopis *et al.* [ML13], the demo[4] of the TV-$L1$ method presented in [Sánchez13a] and the one[5] of Garrido *et al.*[Garrido15] besides IPOL-Brox and *DF-Auto* approaches. We renamed the Horn and Schunck method as IPOL-HS in this experiment.

Here, we observe that the *DF-Auto* and TV-L^1 schemes present an accurate preservation of the motion details, specially *DF-Auto* which solution correctly preserves many details as we observe in figure 5.28. Horn and Schunck achieves a correct result but it also loses many details and makes a huge mistakes. Garrido *et al.* provides a poor solution. These experiments have been made with the corresponding IPOL demos for each method using their default parameters.

[3]http://demo.ipol.im/demo/20/
[4]http://demo.ipol.im/demo/26/
[5]http://demo.ipol.im/demo/112/

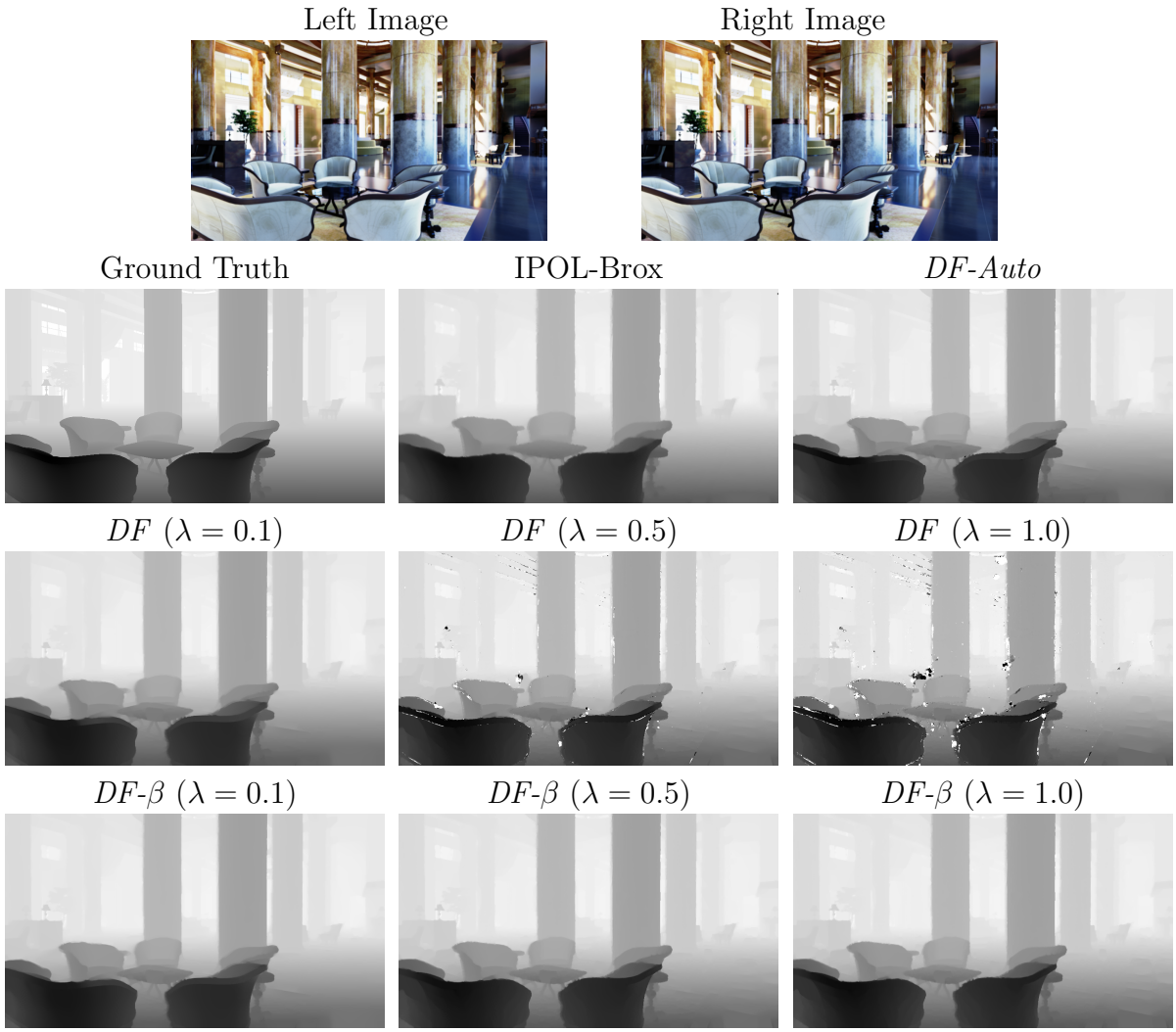


Figure 5.25: *Salon* sequence. First row: Left and right images. Second row: Ground Truth, IPOL-Brox and *DF-Auto* solutions. Third and fourth rows: Flow fields for increasing values of the λ parameter (*DF* and *DF-*β approaches, respectively). The solutions are very accurate, especially at the contours. This accuracy is also present in the IPOL-Brox solution despite of strongest regulariser in the column and some other areas. As expected, the *DF* method is very unstable respect to λ.

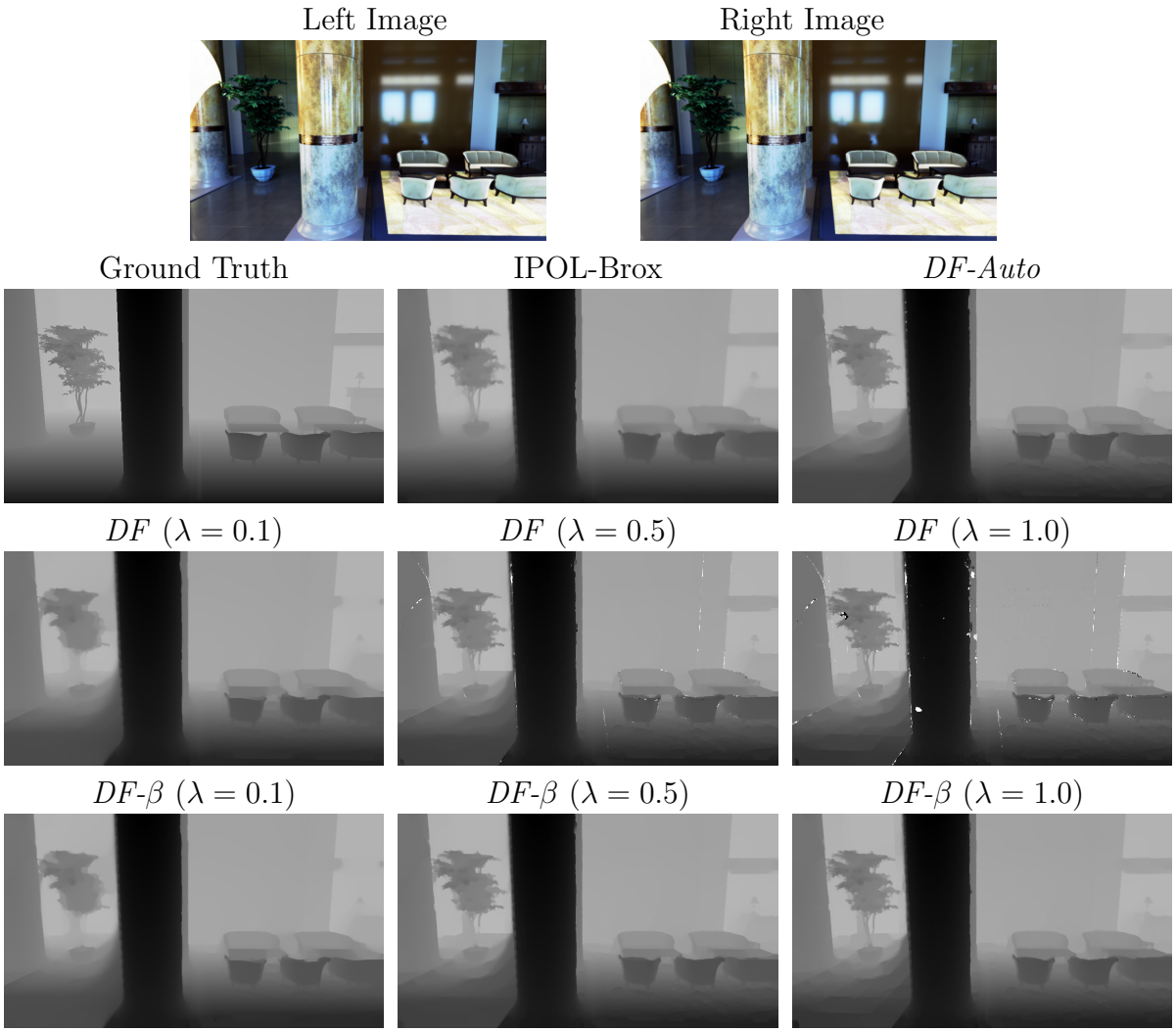


Figure 5.26: *Pillar* sequence. First row: Left and right images. Second row: Ground Truth, IPOL-Brox and *DF-Auto* solutions. Third and fourth rows: Flow fields for increasing values of the λ parameter (*DF* and *DF-*β approaches, respectively). Here, the TV scheme of Brox present an interesting accuracy in the solution and the rounded shapes are not too pronounced as usual. However, the accuracy of the automatic approach is better in the arm-chairs and the lamp at the bottom of the scene.

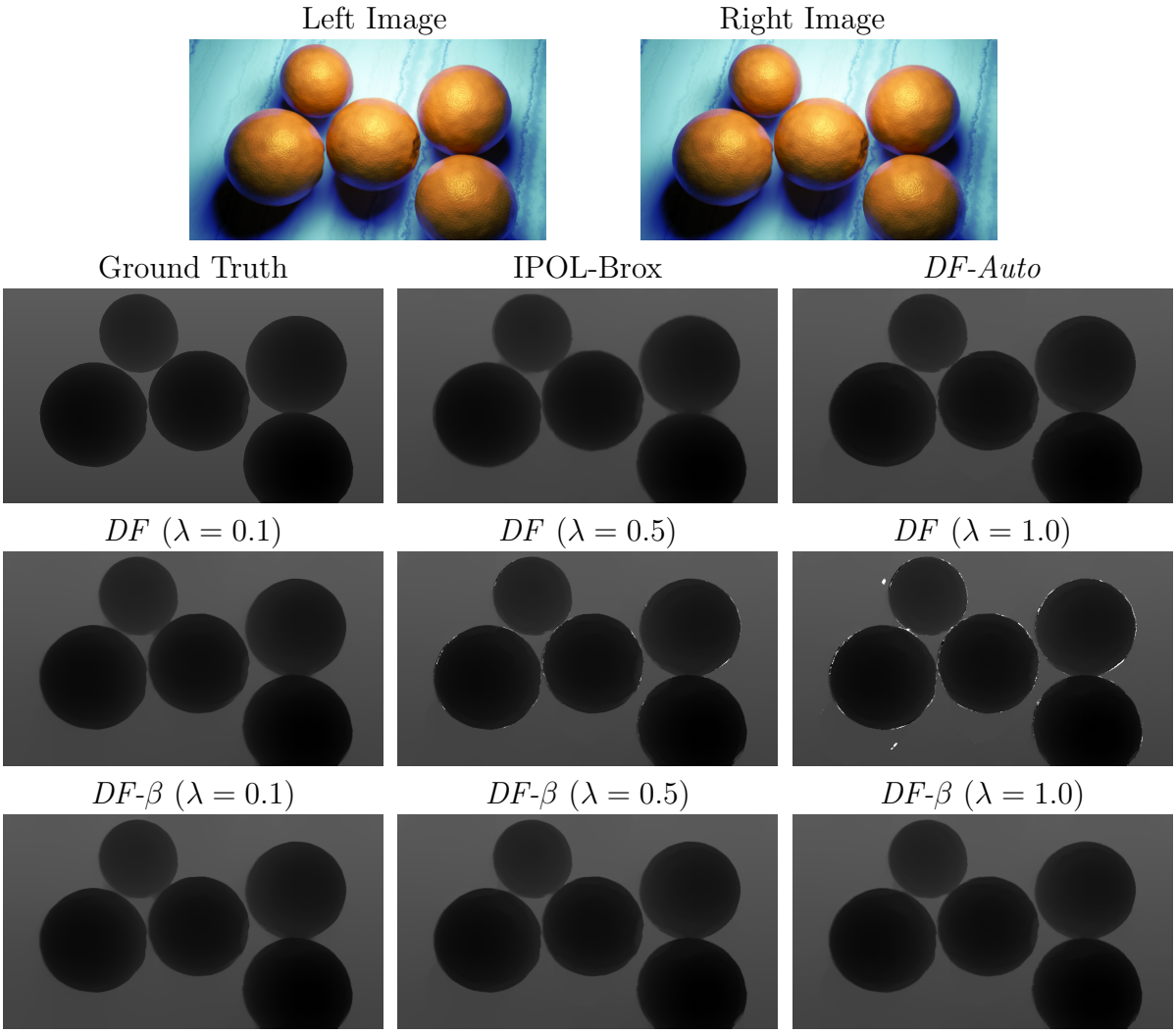


Figure 5.27: *Oranges* sequence. First row: Left and right images. Second row: Ground Truth, IPOL-Brox and *DF-Auto* solutions. Third and fourth rows: Flow fields for increasing values of the λ parameter (*DF* and *DF-*β approaches, respectively). In this scene, all the approaches present good solutions but with a slightly better behavior of the *DF-Auto* proposal.

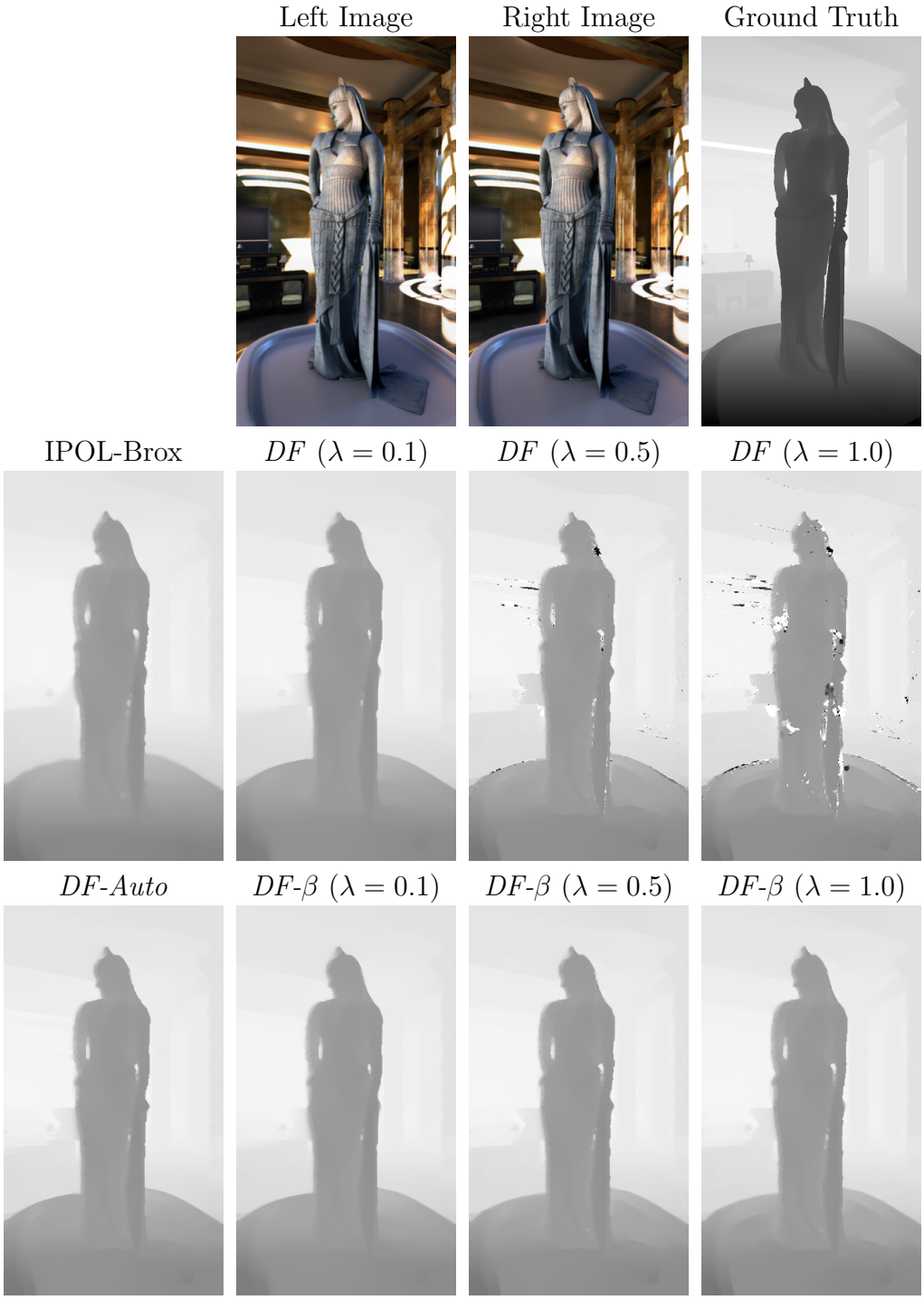


Figure 5.28: *Bastet* sequence. First row: Left and right images and their corresponding Ground Truth. Second row: IPOL-Brox solution and *DF* evolution with respect to λ. Third row: *DF-Auto* solution and results for increasing values of the λ parameter (*DF-*β approach). This figure remarks the superior stability of the new proposals respect to the original exponential method.

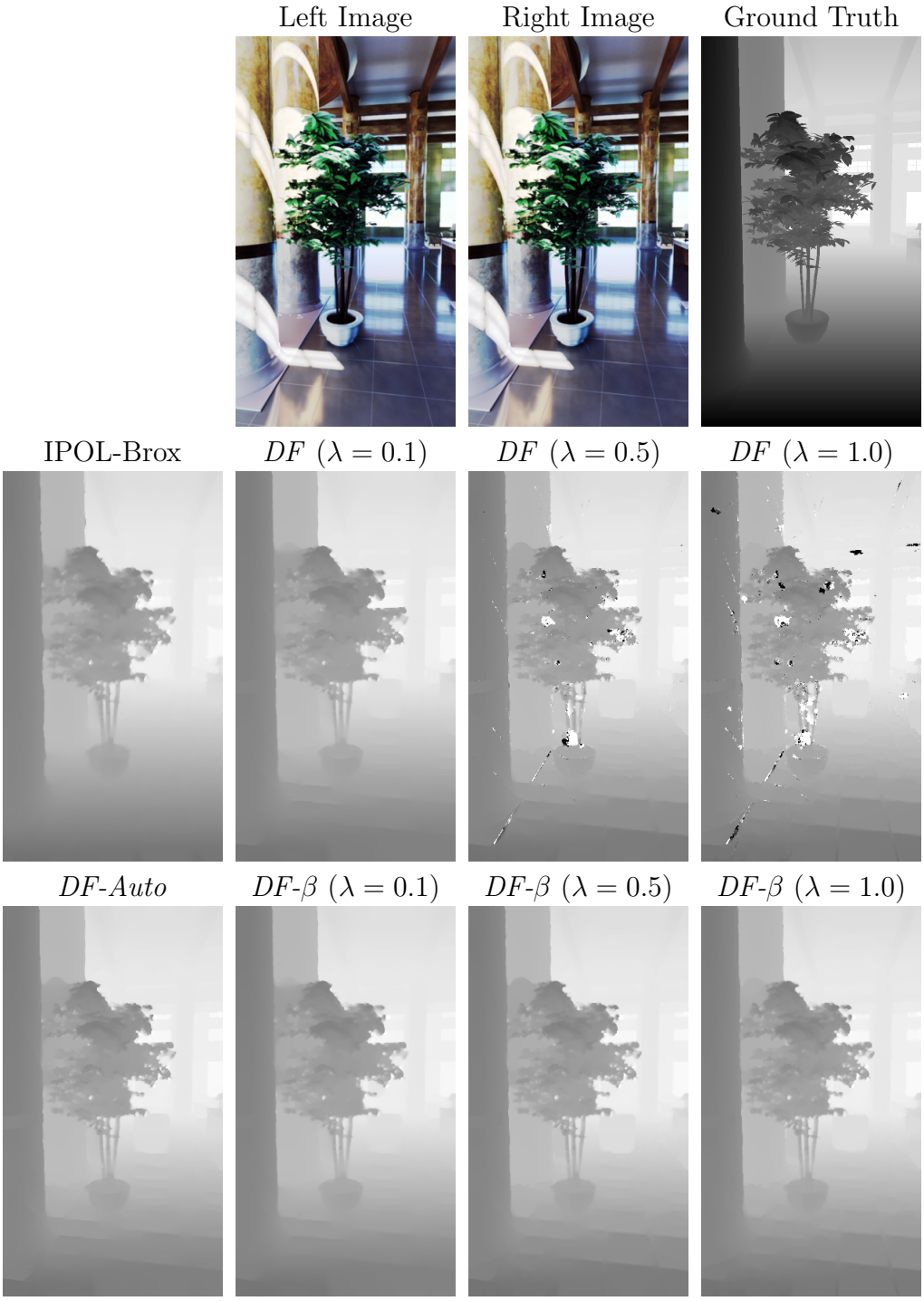


Figure 5.29: *Shrub* sequence. First row: Left and right images and the corresponding Ground Truth. Second row: IPOL-Brox solution and *DF* evolution with respect to λ. Third row: *DF-Auto* solution and results for increasing values of the λ parameter (*DF-β* approach). Here, we observe the *staircasing* problems that generate the exponential functions. The IPOL-Brox regularization is much better in the floor. Nevertheless, the column and the pot are worst conserved compared to the flow achieved by *DF-Auto*.

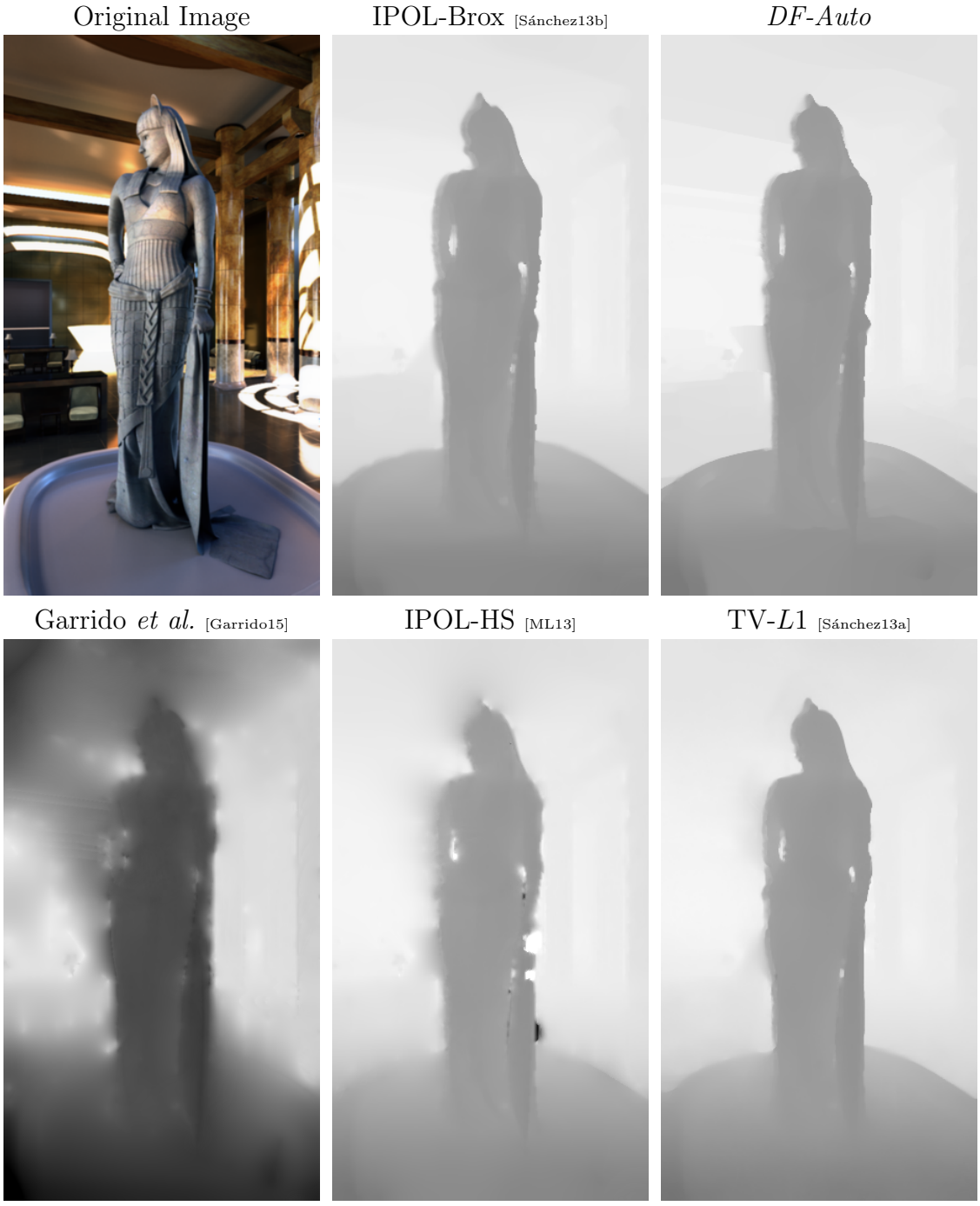


Figure 5.30: Comparison between IPOL-Brox [Sánchez13b], *DF-Auto*, Garrido *et al.* [Garrido15], IPOL-HS [ML13] and TV-$L1$ [Sánchez13a] methods using the *Bastet* sequence.

Table 5.3: Average EPE for the test sequences of *Middlebury* and *Sintel* training datasets

	λ	*Middlebury*	*Sintel-final*	*Sintel-clean*
IPOL-Brox	-	**0.326**	**7.823**	**5.062**
DF	0.1	**0.294**	**7.995**	**5.195**
	0.3	0.326	9.048	6.312
	0.5	0.559	10.467	7.903
DF-β	0.1	0.295	**8.035**	**5.190**
	0.3	**0.292**	8.643	5.823
	0.5	0.312	9.194	6.405
DF-Auto	-	**0.298**	**8.429**	**5.929**

5.4 Numerical Results

Next, we present numerical results using the complete datasets of *Middlebury* and *Sintel*. Table 5.3 shows the average EPE, using different λ values for *DF* and *DF-β*. We observe that the best average error for *Middlebury* is achieved by *DF-β* with $\lambda = 0.3$. Interestingly, the error does not strongly vary for the other λ values. On the other hand, *DF* attains its best result at $\lambda = 0.1$. However, the error considerably increases for the other values. *DF-Auto* scheme obtains a good solution close to the best error. This behavior is similar for the exponential approaches using the *Sintel* dataset. However, IPOL-Brox presents better numerical errors despite of its worst behavior at motion contours (see Fig. 5.32). Once again, we renamed the Brox method as IPOL-Brox because the comparison have been made with our own implementation.

In Table 5.4, we compare our strategies with some of the methods presented in the *Middlebury* ranking. We have chosen the techniques that are somehow related to our approaches. However, note that these are usually much more complex and it is difficult to draw clear conclusions. We see that the results of *DF-β* and *DF-Auto* are better than IPOL-Brox. In Figure 5.31, we show the computed flows for three sequences. We observe that the *DF-Auto* method preserves the contours of the moving objects, even better than *OFH* and *EpicFlow* in many cases. However, we see that the errors at the occlusions are bigger. This is reasonable because the method does not include any mechanism to deal with this problem. Note that we have use the *Middlebury* color scheme in these experiments.

In Tables 5.5 and 5.6, we show the errors for each sequence of the test dataset of *Sintel*, which are published on the web page. We compare with other related methods, such as *EpicFlow*, *LDOF*, *Horn and Schunck* and *IPOL-Brox*. We observe that, as in the previous table, the methods achieve similar results. The proposed methods present competitive results despite of their simplicity. We also notice that the *DF* scheme required much more training to obtain a suitable configuration, whereas *DF-β* and *DF-Auto* did not.

Table 5.4: Best Average EPE for the evaluation sequences of *Middlebury* dataset

	Average EPE	**Rank**
OFH [Zimmer11]	0.360	41.0
EpicFlow [Revaud15]	0.3925	54.3
DF-β	0.4450	68.7
DF-Auto	0.4563	66.9
Brox [Brox04]	0.5013	68.3
TV-L1-improved [Wedel09b]	0.5438	62.7
LDOF [Brox11b]	0.5613	79.2

Table 5.5: Results by sequence on the *MPI Sintel Clean* test subset

	Epic Flow	*LDOF*	*HS*	*IPOL-Brox*	***DF***	***DF-β***	***DF-Auto***
P. Market 3	0.807	1.177	1.450	1.191	1.263	1.263	1.468
P. Shaman 1	0.501	1.612	1.895	1.857	1.649	1.651	1.581
Ambush 1	9.037	34.70	32.97	20.63	22.65	22.69	33.44
Ambush 3	5.657	8.960	10.60	9.233	9.391	9.325	9.880
Bamboo 3	1.030	1.036	1.339	1.106	1.237	1.228	1.344
Cave 3	5.075	7.550	12.41	8.635	8.923	8.929	11.40
Market 1	2.211	3.233	4.650	4.456	4.396	4.392	5.820
Market 4	22.05	38.43	42.94	37.99	38.18	38.06	40.79
Mountain 2	0.225	1.179	0.233	1.025	1.028	1.028	1.094
Temple 1	0.818	1.460	2.056	1.484	1.397	1.395	1.443
Tiger	0.630	1.254	1.064	0.954	1.101	1.097	1.134
Wall	4.330	5.372	6.950	6.554	6.419	6.418	6.375

Table 5.6: Results by sequence on the *MPI Sintel Final* test subset

	Epic Flow	*LDOF*	*HS*	*IPOL-Brox*	***DF***	***DF-β***	***DF-Auto***
P. Market 3	1.290	2.832	2.118	2.057	2.003	2.002	2.080
P. Shaman 1	0.641	2.269	2.463	2.457	2.158	2.159	1.983
Ambush 1	35.18	44.96	40.55	44.92	43.59	43.69	45.50
Ambush 3	8.695	14.13	15.38	14.80	14.32	14.37	15.72
Bamboo 3	1.087	1.107	1.419	1.196	1.355	1.353	1.483
Cave 3	6.207	9.227	13.01	9.704	9.681	9.711	11.31
Market 1	3.150	4.179	5.397	4.809	4.758	4.753	5.401
Market 4	26.55	39.21	40.01	38.26	39.19	39.17	40.79
Mountain 2	1.646	1.618	1.544	1.691	1.767	1.766	1.905
Temple 1	1.295	1.606	2.069	1.741	1.846	1.841	1.909
Tiger	1.147	1.637	1.584	1.799	1.906	1.907	1.987
Wall	4.904	7.294	7.889	7.660	7.594	7.596	7.438

Table 5.7: Results on the *MPI Sintel Clean* test subset

	Rank	**EPE all**	**EPE** matched	**EPE** unmatched	**d0-10**	**s40+**
EpicFlow [Revaud15]	7	4.115	1.360	26.60	3.660	25.86
IPOL-Brox [Sánchez13b]	44	7.283	3.150	40.93	5.705	46.80
DF-β	46	7.391	3.153	41.89	5.492	47.84
DF	47	7.406	3.164	41.94	5.504	47.95
LDOF [Brox11b]	48	7.563	3.432	41.17	5.353	51.70
DF-Auto	53	8.480	3.945	45.40	6.445	56.78
HS [Horn81]	55	8.739	4.525	43.03	7.542	58.24
AnisoHuber[Werlberger09]	61	12.64	7.983	50.47	10.457	77.84

Table 5.8: Results on the *MPI Sintel Final* test subset

	Rank	**EPE all**	**EPE** matched	**EPE** unmatched	**d0-10**	**s40+**
EpicFlow [Revaud15]	9	6.285	3.060	32.56	5.205	38.02
LDOF [Brox11b]	46	9.116	5.037	42.34	6.849	57.30
DF	49	9.188	4.758	45.31	6.821	53.78
DF-β	50	9.196	4.765	45.33	6.829	53.88
IPOL-Brox [Sánchez13b]	51	9.198	4.869	44.48	6.856	53.96
HS [Horn81]	53	9.610	5.419	43.73	7.950	58.27
DF-Auto	54	9.723	5.200	46.59	7.483	57.74
AnisoHuber [Werlberger09]	60	11.93	7.323	49.37	9.464	74.80

In Tables 5.7 and 5.8, we show the global errors. In the *Clean* dataset, the *IPOL-Brox* method provides slightly better results than the strategies with exponential weighting, whereas in the *Final* set occurs the opposite. However, the differences are not meaningful. *DF-Auto* results are slightly worse because of the errors in a few sequences with very large displacements and occlusions, like *Ambush 1*.

Figure 5.32 shows several optical flows. We see that *DF-Auto* detects many details and contours in comparison with the other methods. For instance, in the *Wall* sequence, it preserves the details of the motion at the face and hair, and the flow is not so regularized at the contours. In this case, *EpicFlow* fails to detect the correct motion in two regions. In *Perturbed Shaman 1*, *EpicFlow* provides good results at discontinuities. *DF-Auto* also provides good results, better than *LDOF* and *IPOL-Brox*. With *Temple 1*, it fails to detect the motion of the little dragon. This is due to the problem of small structures with large displacements, for which the pyramidal structure is not suitable.

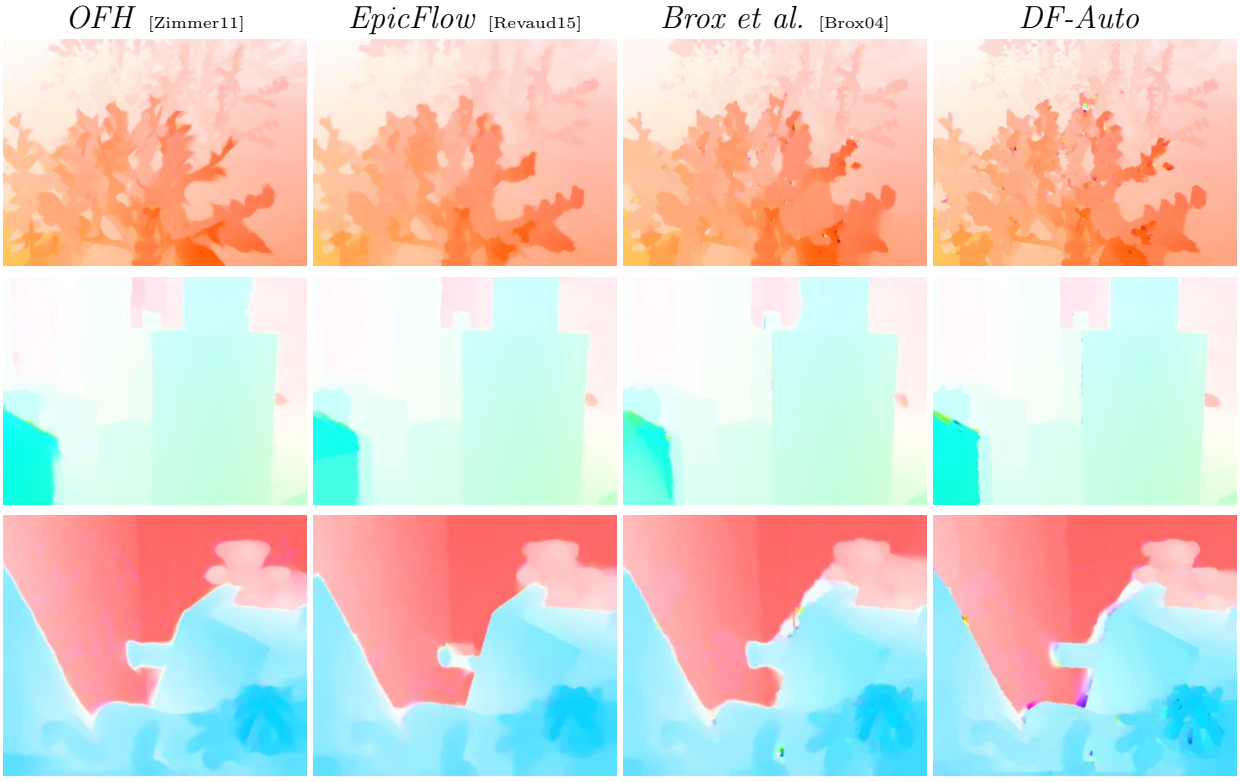


Figure 5.31: Comparison of several optical flow methods using the Middlebury evaluation dataset. From top to bottom: The results obtained with *OFH*, *EpicFlow*, *IPOL-Brox* and *DF-Auto* for Grove, Urban and Teddy, respectively.

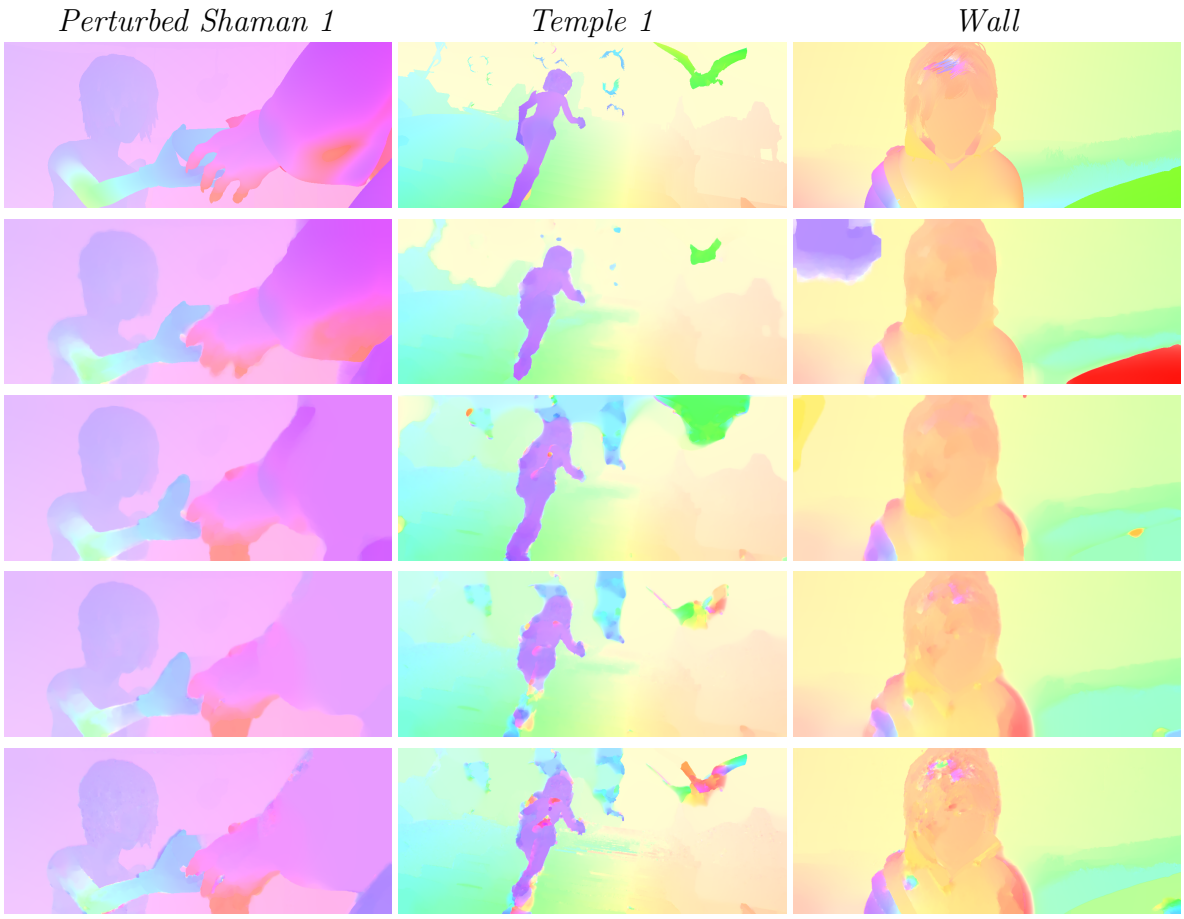


Figure 5.32: Comparison of several optical flow methods using the Sintel Clean test dataset. From top to bottom: The ground truth of three sequences and the results obtained with *EpicFlow* [Revaud15], *LDOF* [Brox11b], *IPOL-Brox* [Sánchez13b] and *DF-Auto*, respectively

5.5 Conclusions

In this chapter, we studied several strategies for preserving discontinuities in TV-L^1 optical flow methods. In particular, we focused our efforts in the use of decreasing functions and anisotropic diffusion tensors. Our experiments showed the benefits and drawbacks of all the strategies. Then, we analyzed the ill-posed problem that may arise when decreasing functions are used in the regularization term.

The *RADT* method is very stable and it does not usually introduced instability over the computed flow field. In this case, the regularization is always present in the isocontour direction and it detects some details that the other methods do not, such as in *Alley 1* or *Grove2*. However, we also notice the motion contours are typically smoother, its accuracy is smaller and it is the most complex approach. Regarding to these, the exponential strategies are very easy to implement from the basic Brox *et al.* method.

We observed that the pure exponential function is highly unstable and demonstrated that the problem of instabilities is not anecdotal, since they appear in any kind of sequence. As consequence, we proposed two efficient techniques to overcome this situation. In the experiments, we compared the strategies and showed the better performance at motion boundaries of our proposals with respect to Brox, *RADT* and *DF* methods.

Our first proposal is very stable and it achieves competitive solutions. In fact, it usually attains the best results for a large range of parameters. Nevertheless, we still need a correct selection of the parameter.

In this sense, the automatic method computes the best parameter and adapts its value to the varying range of image gradients. The experiments showed that, without any user intervention, this approach preserves motion contours while avoiding instabilities. It also provides good results which are usually close to the best solutions. Besides, our study of the gradient parameter (γ) also demonstrated a better behavior and an easier configuration of its value.

From these experiments, we conclude that an exponential strategy offers a precise definition of motion contours only if the parameters are correctly chosen. Otherwise, it could provide poorest results than the Brox approach and may turn unstable.

Interestingly, our two proposals yield good optical flows with more stable solutions for a large range of parameters. Furthermore, we also tested the automatic method with an stereo dataset and observed very accurate results that preserves many details.

Chapter 6

Conclusions and Future Work

6.1 Conclusions

Optical flow is a major theme in the field of computer vision. Despite of more than thirty years of optical flow studies, there are still some limitations in the current methods like the occlusion handling, the large displacements and the preservation of motion boundaries. In this context, the aim of this work was contributing in these problems in two ways:

- The management of the temporal information in optical flow methods.

  In chapter 3, we proposed a variational method that uses temporal coherence for finding more continuous flow fields. This method mixes a nonlinear flow constancy assumption (FCA) in the energy model and a nonlinear flow regularization scheme (FRS) at the PDE level. The first conveniently relates flow fields at different time instants while the second produces continuous flows in time. The approach also includes a robust anisotropic diffusion operator based on the Nagel-Enkelmann one and a technique of *backward flow* to find correspondences back in time. We named the method as TCOF.

  We tested it with synthetic and real sequences to compare the solutions with respect to its spatial counterpart. We observe that, when using real sequences, TCOF allows finding continuous flow fields in time and the noise is reduced with respect to the spatial solution. This is an interesting behavior and shows that the proposed assumptions correctly fit with the standard nonlinear brightness and gradient constancy terms.

  TCOF provides important improvements, specially in the presence of large displacements. The proposed scheme is more general than using the continuous temporal regularization of the flow, with the advantage that it conveniently deals with continuous and non-continuous velocities.

- Efficient strategies for dealing with the discontinuity-preserving problem.

  In chapter 4, we conducted a thorough analysis of the Brox *et al* method. We observed that it produces piecewise-smooth flow fields and it is robust against outliers. However, it also creates rounded shapes at the motion contours and these ones usually do not coincide with the image contours. Then, we introduced the spatial approach into a multichannel framework and evaluated the benefits. We concluded that the improvement is evident and the computational cost does not grow up too much and it is justified.

  The color information improves the motion boundaries but the method still produces rounded shapes. This occurs because the regularization process does not depend on the image information but on the flow field. Introducing a decreasing function in the regularization term is a possible solution of this situation. For this reason, we studied several discontinuity-preserving techniques in chapter 5. In particular, we analyzed the use of decreasing functions and anisotropic diffusion tensors.

  We noticed that this strategy effectively provides accuracy at the flow edges but, if we do not carefully choose the parameters, it considerably increases the errors in the computed motion field. The reason is due to the fact that a wrong parameter introduces instabilities in the form of independent blobs of vectors with large magnitude. Furthermore, we have seen that this big problem occurs in most of the sequences.

  In this sense, we proposed two efficient strategies to overcome this situation. The first strategy combines the decreasing function with a small constant that ensures a minimum isotropic smoothing while the second is a fully automatic approach that adapts the diffusion depending on the histogram of the image gradients. We compared their error evolution with respect to the discontinuity parameter and they correctly reduce the instabilities problems. Besides, both proposals present a good performance at flow discontinuities.

  Our first proposal is very stable and provides very competitive solutions close to the best results. Nevertheless, we think that the automatic approach is better because it normally obtains the best solutions and without any user intervention. In our opinion, this feature make the *DF-Auto* method interesting for real applications.

6.2 Future Work

The different methods described in this thesis achieve good results but there still remain several issues that can be done.

The spatio-temporal method could be more consistent using the redundancy that usually appears between consecutive frames. In some occasions, the information in the

following frames is quite similar. We can use this situation for choosing future frames for creating a better matching criteria.

The problem due to the occlusions is another important issue that can be improved with techniques of backward flow. The major effort should be oriented to improve the filling strategies and removing outliers.

The proposals exposed in chapter 5 improve the detection of flow limits. In this sense, applying these regularization strategies in the temporal method could be a good solution to achieve a more consistent flow. The information of the whole sequence could gradually increase the quality of the flow.

On the other hand, we think that the automatic strategy offers good solutions but it could be improved if we include the β constant in the calculation. This could allow us to choose a smaller τ in $f(\nabla I_1)$ (5.5), so that we may detect more discontinuities. The use of this constant will still ensure a good stability.

This kind of regularization schemes fails to detect all the motion discontinuities if there exists a broad range of image and flow gradients. In future works, we will investigate more reliable ways of combining this information in order to determine the correct flow discontinuities. For instance, we observe *staircasing* problems in some solutions. This is because the decreasing function depends on the image gradient treating some image borders as flow edges. A possible solution may be including the flow information in the function. In this sense, the flow gradient can be helpful to distinguish the motion boundaries from textured zones.

Another issue that we want to address is to develop temporal approaches for video stabilization. The temporal information could be useful for this type of applications.

In this thesis, we used a pyramidal structure to estimate large displacements. However, it is interesting to use other strategies like for instance, nearest neighbors, to obtain a region-based matching method for determining the correspondences between the images. Then, the method is based on a segmentation of the images and an exhaustive search for the plausible displacements.

Appendix I: Estimation of the backward flow

The nonlinear temporal method exposed in chapter 3 requires backward flow for its computation. This can be directly obtained from the forward optical flow when the transformation is bijective. Nevertheless, the presence of occlusions and disocclusions makes this problem difficult. A occluded region occurs when several correspondences arrive to the same position in the target image and disocclusions when no correspondences can be established in one location. Thus, the relation between the information in these cases is not bijective and it is not possible to estimate the inverse map directly.

There is a wide range of methods [Barron94, Zitová03, Maintz98] that requires the estimation of the inverse optical flow. Typically, these approaches compute the correspondences from one image to another, which is normally non congruent with its registration in the opposite direction. There exist many symmetric registration methods that look for congruent solutions in both directions. These symmetric methods have arisen in the fields of optical flow and in non-rigid registration problems in general. The estimation of the inverse optical flow, or the inverse registration map, is necessary in this kind of methods and has shown to provide better results than the one-directional solutions.

Another interesting application is in the field of medical imaging for computing anatomical atlases. Atlases are useful for studying and comparing the anatomy of different patients. In order to compute these atlases, a reference model is registered with respect to several patient models. Then the inverse registrations from the patient models to the atlas are needed for averaging the values and create a more realistic model.

The inverse flow has also been used in methods where it is necessary to track the flow in previous time instants. If we have a large sequence of images, then it is interesting to obtain continuous and congruent flows through the frames. The backward flow allows to search for the correspondences in the previous frames and create spatio-temporal constraints that yield continuous flow fields in time.

Here, we present four algorithms proposed in Sánchez *et al.* [Sánchez15]. These ones use directly the forward optical flow and the image intensities for their solutions. We

assume that the forward flow has been computed and we need to estimate the inverse flow with high precision. These algorithms can be classified in two main groups: (i) flow-based methods that only depend on the input flow field; and (ii) image-based methods that also rely on the information of the images. We will show that the former are efficient in running time and memory requirements, whereas the latter are more accurate and stable. The interpolation strategy is an important issue: vectors are typically given in float precision, so we need to interpolate among several values to get the result in a discrete position. Most state-of-the-art methods use nearest neighbor or linear interpolation. We compare both strategies.

The proposed methods automatically deal with occlusions. In this regard, we may find two possible situations: occlusions due to objects that move faster in the scene, typical in fronto-parallel stereoscopic sequences; or, in the most general case, occlusions produced by any static or moving object. The former situation is interesting in that it only depends on the forward flow, which leads to the aforementioned flow-based methods; see [Sánchez13d]. In a general setting, the flow itself is not enough and we also need to use the images (image-based methods), so that the color information allows us to discriminate between several candidates; see [Sánchez13e].

Disocclusions are more difficult to deal with, since no vector points at these regions. In this case, there is a lack of information and the best we can do is to guess their values from the surrounding estimates. These regions can be filled in a post-processing step. We analyze three filling strategies based on the minimum and average values around the area, and an oriented filling strategy.

I.1 Notation and General Framework

Let $\mathbf{I_1}(\mathbf{x})$ and $\mathbf{I_2}(\mathbf{x})$ be two color images in a sequence, with $\mathbf{x} = (x, y)$, and $\mathbf{w}(\mathbf{x}) = (u(\mathbf{x}), v(\mathbf{x}))$ the vector field that establishes the correspondences between the pixels of the images. $\mathbf{w}(\mathbf{x})$ is said to be *dense* if it is a mapping of every pixel in the first image to the pixels in the second image. We define the backward flow, $\mathbf{w}^*(\mathbf{x}) = (u^*(\mathbf{x}), v^*(\mathbf{x}))$, as the inverse of $\mathbf{w}(\mathbf{x})$. $\mathbf{w}^*(\mathbf{x})$ puts in correspondence the pixels in the second image with the pixels in the first image. The forward and backward flows can be related as:

$$\mathbf{w}(\mathbf{x}) = -\mathbf{w}^*(\mathbf{x} + \mathbf{w}(\mathbf{x})) \;\text{ or }\; \mathbf{w}^*(\mathbf{x}) = -\mathbf{w}(\mathbf{x} + \mathbf{w}^*(\mathbf{x})). \tag{I-1}$$

This relation can be intuitively derived from the graphic depicted in figure I-1. The corresponding positions are given by the warping function, $\mathbf{x}_2 \rightarrow \mathbf{x}_1 + \mathbf{w}(\mathbf{x}_1)$.

Typically, $\mathbf{w}(\mathbf{x})$ is not bijective and it is not possible to estimate $\mathbf{w}^*(\mathbf{x})$ in the whole domain, like in occluded and disoccluded regions.

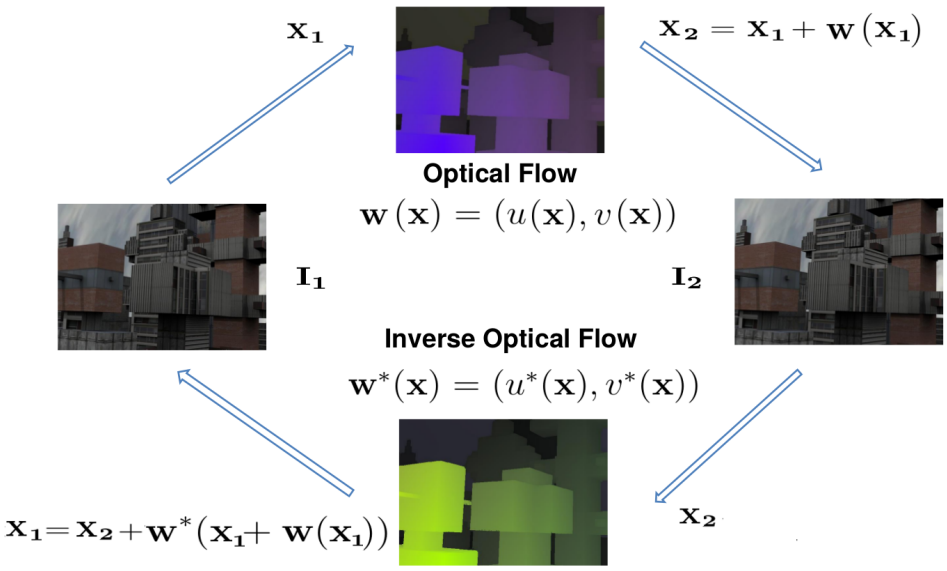


Figure I-1: Relation between the forward, $\mathbf{w}(\mathbf{x})$, and backward, $\mathbf{w}^*(\mathbf{x})$, optical flows.

Since occlusions and disocclusions are common in most natural sequences, $\mathbf{w}(\mathbf{x})$ is in general not invertible. Thus, we have to deal with these two problems in order to find dense inverse functions. These are easy to detect but their solutions have to be overcome from different perspectives. Looking at figure I-2, we observe that occlusions appear in the direction of the moving objects and disocclusions appear in the opposite side.

We also find two possible situations: occlusions may appear when moving objects occlude other background objects, like depicted in figure I-2; on the other hand, occlusions may also occur when objects displace behind other objects, like in figure I-3. The former situation is typical in fronto-parallel stereoscopic sequences, so we will refer this situation as the *stereoscopic occlusion*. It can be efficiently solved, like proposed in [Sánchez13d]. The latter usually appears in general optical flow problems [Sánchez13e].

We call this last situation the *street-lamp occlusion*, as it typically appears in traffic sequences where a car moves behind a street lamp. Looking at figure I-3, we observe that occlusions are not only present in front of the square but also inside it, since these pixels fall exactly behind the bar in the second image. The effect of this occlusion is like projecting the bar into the square at a distant equivalent to its motion.

The problem of occlusions can be solved by means of a selection process. Several vectors point at the same position, and we need to select the appropriate one. The selection problem may only depend on the vector field itself, as it happens in the stereoscopic occlusion. In this case, we can select the value corresponding to the biggest motion. On the other hand, the more general case of the street-lamp occlusion requires

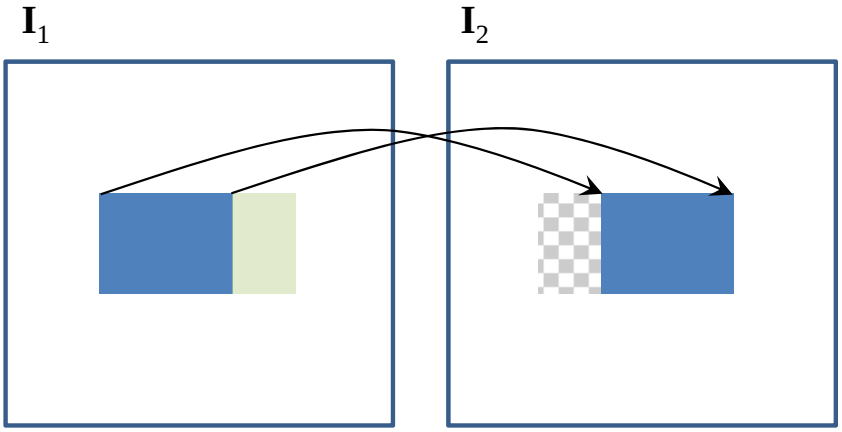


Figure I-2: *Stereoscopic Occlusion*: when the blue square moves horizontally, it creates a disocclusion and occlusion before and after the square, respectively.

not only the values of the optical flow but also the image intensities: we have to select the motion associated with the pixel that is finally visible in the second image.

In the disocclusion problem, there is no information available for a given position. This can be addressed as a filling procedure, selecting the information on the neighborhood of the pixel. Notice the symmetric behavior between occlusions and disocclusions from figures I-2 and I-3.

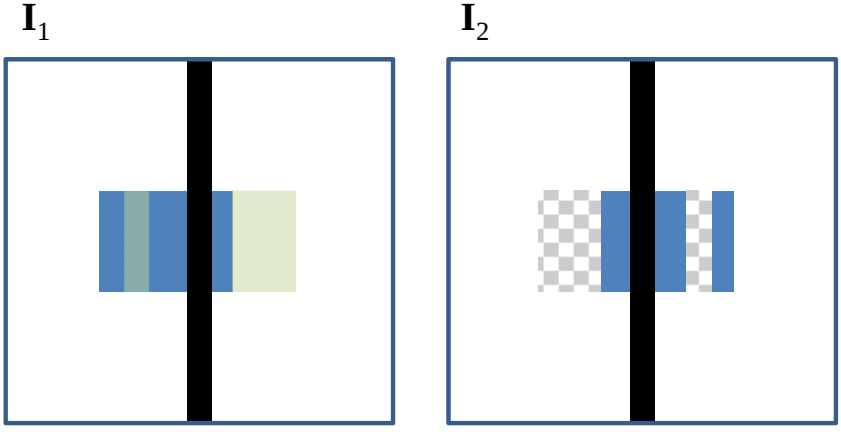


Figure I-3: *Street-lamp Occlusion*: the blue square moves behind the static central bar. Occlusions appear in front of the square and also inside the square, induced by the static bar.

Figure I-4 shows two examples of inverse optical flows. The motion fields are shown using the IPOL color scheme. In this case, we have used two sequences from the

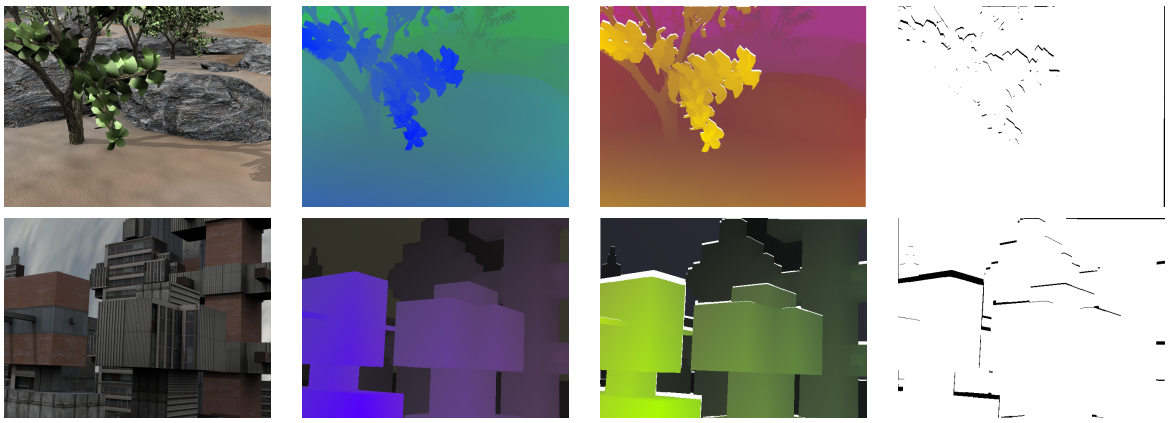


Figure I-4: Two examples using sequences from the Middlebury benchmark database: the first column shows the Grove2 and Urban3 sequences, respectively; the second depicts the forward optical flows; the third, the inverse optical flows; and the last one, the corresponding disocclusions.

Middlebury benchmark database. The forward optical flows are known (second column) and the inverse flows are calculated using one of our algorithms (third column). We also show the disocclusion maps in the fourth column.

In the following section we explain the algorithms. These algorithms have been grouped in two categories: *nearest neighbor algorithms* that use a single value from the forward flow; and *interpolation algorithms* that compute an average between the values that get close to a position. These are explained in subsections I.1.1 and I.1.2, respectively. Nearest neighbor algorithms only need one image pass, while the others need two passes.

These categories are further divided in flow- and image-based strategies. The flow-based strategy uses the information of the forward optical flow, whereas the image-based strategy also uses the information of the image intensities. The former can be applied in the stereoscopic occlusion case and the latter in the more general street-lamp occlusion.

I.1.1 Nearest Neighbor Algorithms

Our first algorithm is based on the optical flow information. It is highly efficient and provides good results for the stereoscopic occlusion case. Algorithm 1 depicts the steps to compute the backward flow. This algorithm is very fast since only one pass through the image is necessary. It does not use any temporal buffer because the operations are carried out in the output backward flow.

$|\cdot|$ stands for the Euclidean norm of a vector. At the beginning, we initialize the inverse flow to zero. In each pixel, $\mathbf{x} = (x, y)$, we compute the corresponding position in the other image, $\mathbf{x} + \mathbf{w}(\mathbf{x})$. Normally, this position will lie in the middle of four pixels.

We obtain the neighbors around this position, given by $\{\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \mathbf{x}_4\}$, and the interpolation weights, w_1, w_2, w_3, w_4, shown in figure I-5. Then, we estimate the magnitude of the forward flow and compare it with the magnitude of the already stored backward flows.

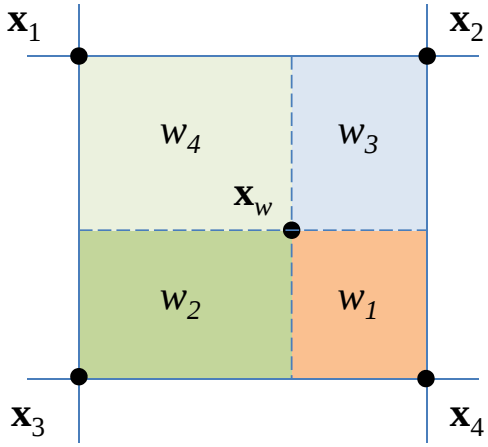


Figure I-5: Weights.

If the value of the forward magnitude is bigger than the previous stored value, and the corresponding weight is bigger than a given threshold, then we keep the negative value of the flow at that position. This threshold has been set to 0,25 because it represents the situation when the correspondence falls exactly in the middle of the pixel. Another alternative is to use distances instead of areas. Nevertheless, the behavior is very similar, but we will prefer areas for the interpolation algorithms.

In this way, occlusions are automatically handled by the algorithm: if there are collisions in one position, we retain the flow with higher magnitude. In the last step, we fill disocclusions, which correspond to the positions that have not been visited in the backward flow.

If we want to deal with the street-lamp occlusion case, then we need more information than the optical flow. Now the objects with the smallest motions can occlude other fast moving objects. Thus, the magnitude of the flow cannot be a discriminant.

We use the intensities of the images as discriminant. We select the motion corresponding to the pixel with the most similar intensities in both images. Algorithm 2 shows the steps of the new method. This algorithm is also efficient and applies to any configuration. Additionally, we need a buffer to store the pixel similarities, so that, in case of an occlusion, we retain the value corresponding to the most similar value.

The input data includes the forward flow and both images. At the beginning, we initialize the buffer to a big number. Then, the algorithm goes over each position of the vector field, carrying out the same steps as in the previous algorithm. The main difference

is that it estimates the similarity between the images at the given positions. It assigns the negative value of the original flow to each of the neighbors, if it has the best similarity and lies close to the destiny pixel.

Notice that similarities are stored in a buffer, so that when there are several pixels that arrive to the same position, i.e., in the case of occlusions, the algorithm retains the value corresponding to the most similar pixels in both images. In this case, the use of the buffer allows us to automatically deal with occlusions. This algorithm is also very fast, since only one pass through the image is necessary.

For the similarity measure, we have used the RGB color information. This may pose some drawbacks if there are brightness changes, which is typical in real sequences. Other robust similarity measures can be used, based on the gradient or the curvature of the images. The noise of the images may also affect the similarity measure. A simple approach to mitigate this problem is to convolve the images with a Gaussian kernel.

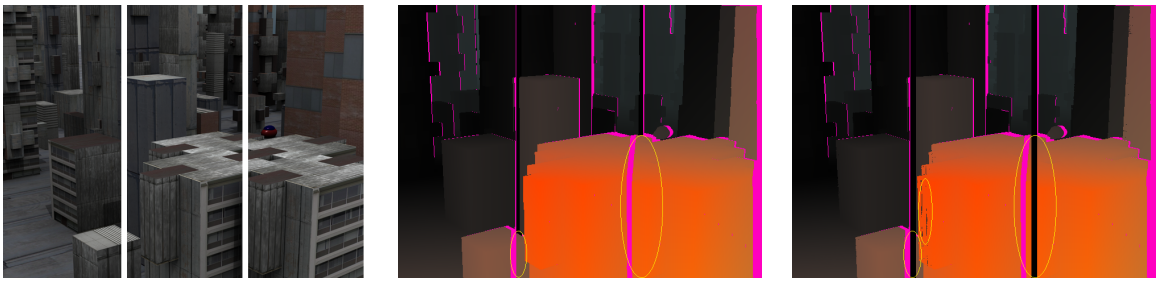


Figure I-6: Dealing with street-lamp occlusions: left, the Urban2 sequence; middle, result of Algorithm 1; and right, result of Algorithm 2. The first algorithm does not detect the motion of the bars and the second algorithm introduces some errors due to brightness changes (see yellow ellipses).

Figure I-6 shows an example of the street-lamp occlusion case. We have modified the Urban2 sequence from the Middlebury benchmark database by introducing two static bars. We show the results for Algorithms 1 and 2. We observe that the first algorithm can not correctly deal with the static bars. On the other hand, the second algorithm can cope with this situation, but it does not detect the correct motion in a few pixels of the front building. This region coincides with an occlusion and the colors are very similar in several regions.

I.1.2 Interpolation Algorithms

Given the discrete nature of images, normally the correspondences fall among several pixels in the other image. In fact, many correspondences may get around one pixel. It could be more convenient to use all this information and compute an average between

these values. In the previous algorithms, we have selected one value; therefore we are introducing a shift on the positions of the inverse flow.

In this section we propose two algorithms addressing this problem. We note that, although it is interesting to compute an average between the correspondences that get around a pixel, at flow discontinuities or occlusions we have to average only with the values of their corresponding regions. This is necessary to avoid degrading these areas.

In order to compute the inverse flow, we calculate the weighted average of the flows, using the following equation:

$$\mathbf{w}^*(\mathbf{x}) = -\frac{\sum\limits_{\mathbf{x}_i \in \mathcal{N}_i} w_i \mathbf{w}(\mathbf{x}_i)}{\sum\limits_{\mathbf{x}_i \in \mathcal{N}_i} w_i}, \tag{I-2}$$

where $\mathcal{N}_i \equiv \{\mathbf{x}_i : |\mathbf{x}_i + \mathbf{w}(\mathbf{x}_i) - \mathbf{x}| < 1\}$, i.e., $\mathcal{N}_i$ is the set of correspondences that fall around $\mathbf{x}$.

The first algorithm in this section relies on a procedure that is in charge of computing the weighted average (I-2). This formula is computed iteratively, as we find new values around a position.

This procedure receives the forward optical flow, $\mathbf{w}$, its magnitude, d, the weight as depicted in figure I-5, $wght$, and the values stored at this position d^*, $\mathbf{w}^*$ and $wght^*$. If the current flow magnitude, d, is within a threshold (`MOTION_TH`) of the stored flow magnitude, then we accumulate the values for $\mathbf{w}^*$ and $wght^*$, in order to progressively estimate (I-2). Otherwise, if the current magnitude is bigger than the stored one, then we initialize these variables to the current values. In this procedure, we use a constant, `MOTION_TH`, that is used to detect similar motions. By default, its value is equal to $0,25$. The second condition allows us to preserve the flow discontinuities and select the correct value at occlusions.

Algorithm 3 is an improvement with respect to Algorithm 1. The steps of both algorithms are essentially the same, with the following differences: the new method uses the buffers **avg_h** and $wght^*$ to accumulate the value of the additions in (I-2); an additional buffer, d^*, to store the calculated flow magnitudes; and four calls to the previous procedure, to select the correct motion in each position. Finally, the algorithm calculates (I-2) using the accumulation buffers. This is carried out through another image pass.

This algorithm is also efficient. It uses two image passes to compute the inverse flow: one for creating the intermediate buffers and another for calculating (I-2). The number of operations in the second pass reduces to two divisions and two assignments per pixel. The main advantage is that the accuracy of the calculated inverse flows is higher.

The fourth algorithm relies on another procedure that plays the same role as the previous one. If the magnitudes of the current and stored flows are similar, then

the procedure accumulates the information. Otherwise, the procedure keeps the flow associated with the most similar pixel intensity. Thus, the main difference is that the discriminant for selecting a value depends on the RGB similarity.

Algorithm 4 computes flow averages and uses the image intensities to select the correct flow. This new method uses the buffers, **avg_h** and $wght^*$ to accumulate the value of the summations in (I-2); an additional buffer, dI, to store the calculated RGB differences of both images; and four calls to the previous procedure, to select the correct motion in each position. A final image pass is necessary to calculate (I-2).

The performance of this algorithm is similar to the previous algorithm. The benefit is that it properly works for stereoscopic and street-lamp occlusions. Nevertheless, it carries out some more operations for calculating the RGB differences between the images. Note that these interpolation algorithms can also cope with occlusions and respect the flow discontinuities.

The algorithms explained in this section can be easily extended to higher order dimensions. This is useful, for instance, in three dimensional medical images. In this case, the problem of occlusions and disocclusions is not important and the accuracy of these methods can be even higher.

Algorithm 1: Inverse Flow

Input: $\mathbf{w}$
Output: $\mathbf{w}^*$

Initialize $\mathbf{w}^*$ to 0
foreach *position* $\mathbf{x}$ **do**
  $\mathbf{x}_w \leftarrow \mathbf{x} + \mathbf{w}(\mathbf{x})$
  Find the four neighbors of $\mathbf{x}_w$: $\{\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \mathbf{x}_4\}$
  Compute the bilinear interpolation weights: w_1, w_2, w_3, w_4 (see figure I-5)
  $d \leftarrow |\mathbf{w}(\mathbf{x})|^2$
  $d_1 \leftarrow |\mathbf{w}^*(\mathbf{x}_1)|^2$
  $d_2 \leftarrow |\mathbf{w}^*(\mathbf{x}_2)|^2$
  $d_3 \leftarrow |\mathbf{w}^*(\mathbf{x}_3)|^2$
  $d_4 \leftarrow |\mathbf{w}^*(\mathbf{x}_4)|^2$
  if $w_1 \geq 0,25$ ***and*** $d \geq d_1$ **then**
    $\mathbf{w}^*(\mathbf{x}_1) \leftarrow -\mathbf{w}(\mathbf{x})$
  end
  if $w_2 \geq 0,25$ ***and*** $d \geq d_2$ **then**
    $\mathbf{w}^*(\mathbf{x}_2) \leftarrow -\mathbf{w}(\mathbf{x})$
  end
  if $w_3 \geq 0,25$ ***and*** $d \geq d_3$ **then**
    $\mathbf{w}^*(\mathbf{x}_3) \leftarrow -\mathbf{w}(\mathbf{x})$
  end
  if $w_4 \geq 0,25$ ***and*** $d \geq d_4$ **then**
    $\mathbf{w}^*(\mathbf{x}_4) \leftarrow -\mathbf{w}(\mathbf{x})$
  end
end
`FillDisocclusions`

Algorithm 2: Inverse Image-based Maximum Flow

Input: $\mathbf{I}_1, \mathbf{I}_2, \mathbf{w}$
Output: $\mathbf{w}^*$

Initialize each position of a buffer to a big number
foreach *position* $\mathbf{x}$ **do**
  $\mathbf{x}_w \leftarrow \mathbf{x} + \mathbf{w}(\mathbf{x})$
  Find the four neighbors of $\mathbf{x}_w : \{\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \mathbf{x}_4\}$
  Compute the bilinear interpolation weights: w_1, w_2, w_3, w_4 (see figure I-5)
  $d_1 \leftarrow |\mathbf{I}_1(\mathbf{x}) - \mathbf{I}_2(\mathbf{x}_1)|^2$
  $d_2 \leftarrow |\mathbf{I}_1(\mathbf{x}) - \mathbf{I}_2(\mathbf{x}_2)|^2$
  $d_3 \leftarrow |\mathbf{I}_1(\mathbf{x}) - \mathbf{I}_2(\mathbf{x}_3)|^2$
  $d_4 \leftarrow |\mathbf{I}_1(\mathbf{x}) - \mathbf{I}_2(\mathbf{x}_4)|^2$
  if $w_1 \geq 0,25$ ***and*** $buffer(\mathbf{x}_1) \geq d_1$ **then**
    $\mathbf{w}^*(\mathbf{x}_1) \leftarrow -\mathbf{w}(\mathbf{x})$
    $buffer(\mathbf{x}_1) \leftarrow d_1$
  end
  if $w_2 \geq 0,25$ ***and*** $buffer(\mathbf{x}_2) \geq d_2$ **then**
    $\mathbf{w}^*(\mathbf{x}_2) \leftarrow -\mathbf{w}(\mathbf{x})$
    $buffer(\mathbf{x}_2) \leftarrow d_2$
  end
  if $w_3 \geq 0,25$ ***and*** $buffer(\mathbf{x}_3) \geq d_3$ **then**
    $\mathbf{w}^*(\mathbf{x}_3) \leftarrow -\mathbf{w}(\mathbf{x})$
    $buffer(\mathbf{x}_3) \leftarrow d_3$
  end
  if $w_4 \geq 0,25$ ***and*** $buffer(\mathbf{x}_4) \geq d_4$ **then**
    $\mathbf{w}^*(\mathbf{x}_4) \leftarrow -\mathbf{w}(\mathbf{x})$
    $buffer(\mathbf{x}_4) \leftarrow d_4$
  end
end
`FillDisocclusions`

Procedure SelectMotion($d, \mathbf{w}, wght, d^*, \mathbf{w}^*, wght^*$)

if $wght \geq 0,25$ **then**
  if `abs`$(d - d^*) \leq$ `MOTION_TH` **then**
    $\mathbf{w}^* \leftarrow \mathbf{w}^* + \mathbf{w} * wght$
    $wght^* \leftarrow wght^* + wght$
  else if $d \geq d^*$ **then**
    $d^* \leftarrow d$
    $\mathbf{w}^* \leftarrow \mathbf{w} * wght$
    $wght^* \leftarrow wght$
  end
end

Algorithm 3: Inverse Average Flow

Input: $\mathbf{w}$
Output: $\mathbf{w}^*$

Initialize buffers $d^*, \mathbf{avg\_h}, wght^*$ to 0
foreach *position* $\mathbf{x}$ **do**
  $\mathbf{x}_w \leftarrow \mathbf{x} + \mathbf{w}(\mathbf{x})$
  Find the four neighbors of $\mathbf{x}_w : \{\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \mathbf{x}_4\}$
  Compute the bilinear interpolation weights: w_1, w_2, w_3, w_4 (see figure I-5)
  $d \leftarrow |\mathbf{w}(\mathbf{x})|^2$
  `SelectMotion` $(d, \mathbf{w}(\mathbf{x}), w_1, d^*(\mathbf{x}_1), \mathbf{avg\_h}(\mathbf{x}_1), wght^*(\mathbf{x}_1))$
  `SelectMotion` $(d, \mathbf{w}(\mathbf{x}), w_2, d^*(\mathbf{x}_2), \mathbf{avg\_h}(\mathbf{x}_2), wght^*(\mathbf{x}_2))$
  `SelectMotion` $(d, \mathbf{w}(\mathbf{x}), w_3, d^*(\mathbf{x}_3), \mathbf{avg\_h}(\mathbf{x}_3), wght^*(\mathbf{x}_3))$
  `SelectMotion` $(d, \mathbf{w}(\mathbf{x}), w_4, d^*(\mathbf{x}_4), \mathbf{avg\_h}(\mathbf{x}_4), wght^*(\mathbf{x}_4))$
end
foreach *position* $\mathbf{x}$ **do**
  if *pixel* $\mathbf{x}$ *is not a disocclusion* **then**
    $\mathbf{w}^*(\mathbf{x}) \leftarrow -\frac{\mathbf{avg\_h}(\mathbf{x})}{wght^*(\mathbf{x})}$
  end
end
`FillDisocclusions`

Procedure SelectImageMotion($dI, \mathbf{w}, wght, dI^*, \mathbf{w}^*, wght^*$)

```
1  if wght ≥ 0,25 then
2  |  d ← |w|^2
3  |  if abs(d − d*) ≤ MOTION_TH then
4  |  |  w* ← w* + w · wght
5  |  |  wght* ← wght* + wght
6  |  else if dI* ≥ dI then
7  |  |  d* ← d
8  |  |  dI* ← dI
9  |  |  w* ← w · wght
10 |  |  wght* ← wght
11 |  end
12 end
```

Algorithm 4: Inverse Image-based Average Flow

Input: $\mathbf{I}_1, \mathbf{I}_2, \mathbf{w}$
Output: $\mathbf{w}^*$

Initialize buffers $\mathbf{avg\_h}, wght^*$ to 0
Initialize buffer dI to a big number
foreach *position* $\mathbf{x}$ **do**
  $\mathbf{x}_w \leftarrow \mathbf{x} + \mathbf{w}(\mathbf{x})$
  Find the four neighbors of $\mathbf{x}_w : \{\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \mathbf{x}_4\}$
  Compute the bilinear interpolation weights: w_1, w_2, w_3, w_4 (see figure I-5)
  $d_1 \leftarrow |\mathbf{I}_1(\mathbf{x}) - \mathbf{I}_2(\mathbf{x}_1)|^2$
  $d_2 \leftarrow |\mathbf{I}_1(\mathbf{x}) - \mathbf{I}_2(\mathbf{x}_2)|^2$
  $d_3 \leftarrow |\mathbf{I}_1(\mathbf{x}) - \mathbf{I}_2(\mathbf{x}_3)|^2$
  $d_4 \leftarrow |\mathbf{I}_1(\mathbf{x}) - \mathbf{I}_2(\mathbf{x}_4)|^2$
  `SelectImageMotion` $(d_1, \mathbf{w}(\mathbf{x}), w_1, dI(\mathbf{x}_1), \mathbf{avg\_h}(\mathbf{x}_1), wght^*(\mathbf{x}_1))$
  `SelectImageMotion` $(d_2, \mathbf{w}(\mathbf{x}), w_2, dI(\mathbf{x}_2), \mathbf{avg\_h}(\mathbf{x}_2), wght^*(\mathbf{x}_2))$
  `SelectImageMotion` $(d_3, \mathbf{w}(\mathbf{x}), w_3, dI(\mathbf{x}_3), \mathbf{avg\_h}(\mathbf{x}_3), wght^*(\mathbf{x}_3))$
  `SelectImageMotion` $(d_4, \mathbf{w}(\mathbf{x}), w_4, dI(\mathbf{x}_4), \mathbf{avg\_h}(\mathbf{x}_4), wght^*(\mathbf{x}_4))$
end
foreach *position* $\mathbf{x}$ **do**
  if *pixel* $\mathbf{x}$ *is not a disocclusion* **then**
    $\mathbf{w}^*(\mathbf{x}) \leftarrow -\frac{\mathbf{avg\_h}(\mathbf{x})}{wght^*(\mathbf{x})}$
  end
end
`FillDisocclusions`

I.1.3 Filling Disocclusions

The last step of the algorithms is to fill disocclusions. These are regions with no correspondences in the source image, so we get empty spaces in the backward flow. It is not possible to find an exact solution if only two images are taken into account. Probably, the use of more frames may allow us to uniquely identify the values inside these regions.

The best we can do is to guess the information from the neighbor values. We propose three filling strategies: on the one hand, we use a minimum fill strategy that looks for the minimum value around the disocclusion; then, we propose an averaging solution that computes the average from the values around the region; finally, we propose an oriented filling scheme that searches the value in the opposite direction to the flow.

I.1.4 Minimum Fill Strategy

The min-fill strategy is justified in the case of stereoscopic occlusions. In this situation, disocclusions are associated with the lowest moving objects, so it seems reasonable to select the minimum motion. In [Sánchez13d], we analyzed two min-fill strategies.

The first strategy uses a connected component labeling process to group the regions. It assigns a label to each region, finds a unique minimum around each area, and then assigns the value to each position inside. This strategy is simple and fast. It works correctly if the size of the regions is small. However, when regions are large, it may assign values which are far from the position.

The second strategy finds the minimum value that is near the current position. We go through the image and try to fill disocclusions using a fix-sized window. The size of the window may not be large enough to attain values outside the region. In this case, the process is run again to fill the remaining disocclusions. The default window radius is 5 in the experimental results. In the experimental results, we use this strategy, since it usually provides better results, as shown in [Sánchez13d].

I.1.5 Average Fill Strategy

An alternative to the min-fill strategy is to use an average filling. Instead of selecting a unique value, we compute an average in a fix-sized window. The process is similar to the previous approach: we accumulate the values and assign the average to its corresponding position if the number of values is greater than a threshold. This threshold is equal to the windows radius (5 in the experiments). This process is carried out iteratively until every disocclusion is successfully filled.

This approach is a compromise between both types of occlusions: it is not going to work as well as the min-fill strategy for the stereoscopic case, but it should outperform this strategy for the street-lamp occlusion.

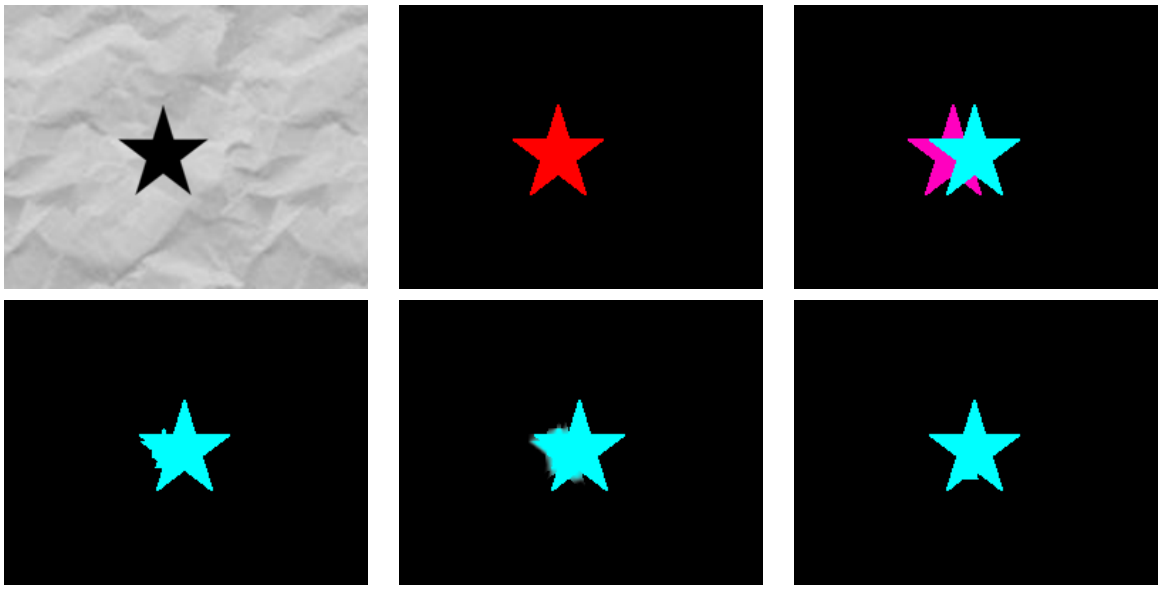


Figure I-7: Comparison of filling strategies. In the first row: first column, the left image; second, the true flow; and third, the inverse flow with disocclusions in pink. In the second row: first column, the solution using the min-fill strategy; second, using the average-fill strategy; and third, using the oriented-fill strategy.

I.1.6 Oriented Fill Strategy

We propose another strategy that takes into account the information of the flow orientation. If one object moves in one direction, it seems reasonable that the motion at disocclusions corresponds to the motion that is in the opposite direction.

In this sense, what we do is the following: we first compute the search vector for each position as $-\frac{\mathbf{w}(\mathbf{x})}{|\mathbf{w}(\mathbf{x})|}$; then, it iterates until it finds a value outside the disoccluded region in this direction.

Note that we use the forward flow to estimate the vector orientation. This probably works well for the stereoscopic occlusion, since normally the motion corresponds to the backward object. In the street-lamp case, this strategy works properly if the disocclusion falls inside the correct object.

An improvement to this strategy is to use, not only one vector in the opposite direction, but also its orthogonal vectors, so that the filling process searches for information in more directions. This can be efficiently addressed through an oriented diffusion process.

In figure I-7 we compare these filling strategies. We use a simple sequence of a star moving fifteen pixels horizontally. This sequence is interesting because disoccluded regions are large and complex (see the image on the first row and last column). The performance of the average-fill strategy (second row in the middle) is the poorest. The min-fill strategy provides good results, but when the search window is not large enough, then it takes values from the motion of the star. Finally, the oriented-fill strategy provides the best results in this case.

I.1.7 Analysis of the Inverse Optical Flow Algorithms

In order to evaluate the methods, we use an error measure based on the following *reprojection error*: we compute the backward flow twice, $(\mathbf{w}^*(\mathbf{x}))^*$; then, we compare the result with the original true flow using the average End-point Error (EPE) and Angular Error (AAE), as explained in [Baker07b]. figure I-8 shows the three functions – $\mathbf{w}(\mathbf{x})$, $\mathbf{w}^*(\mathbf{x})$ and $(\mathbf{w}^*(\mathbf{x}))^*$– for the Urban2 sequence. The last two functions have been computed using Algorithm 1 and the min-fill strategy.

Notice that $(\mathbf{w}^*(\mathbf{x}))^*$ is slightly different than $\mathbf{w}(\mathbf{x})$, especially at occlusions. In the experimental results we will be using the EPE and AAE between these two functions for comparing the accuracy of the algorithms. We will use the test sequences from the Middlebury benchmark database [Baker07b], which provides the correct ground truths. Unfortunately, there are some sequences in the database that mix the information of occlusions with the optical flows, so we will only use these when possible.

Another alternative is to use a symmetric error metric based on (I-1) as $|\mathbf{w}(\mathbf{x}) + \mathbf{w}^*(\mathbf{x} + \mathbf{w}(\mathbf{x}))|$. This has been previously used in the symmetrical optical flow method explained in [Alvarez07]. However, we need to use an interpolation method to estimate $\mathbf{w}^*(\mathbf{x} + \mathbf{w}(\mathbf{x}))$, which will introduce a bias in the computed errors, especially at discontinuities. The advantage of the previous metric is that it only depends on the inverse calculations.

I.1.8 Analysis of Inverse Flow Accuracy

In this section we analyze the methods without taking into account disocclusions. Figures I-9 and I-10 show the results for Algorithm 1.

Disocclusions are marked in pink. We can see that the precision of the inverse optical flow seems high and occlusions are correctly handled by the algorithm. Disocclusions are large in some sequences like, for instance, in the building or tree sequences.

The EPE and AAE results are shown in Tables I-1 and I-2, respectively. The best result per row is in bold letters and the second in italics. The first algorithm provides good results but these are the poorest in the table. This method is interesting because of

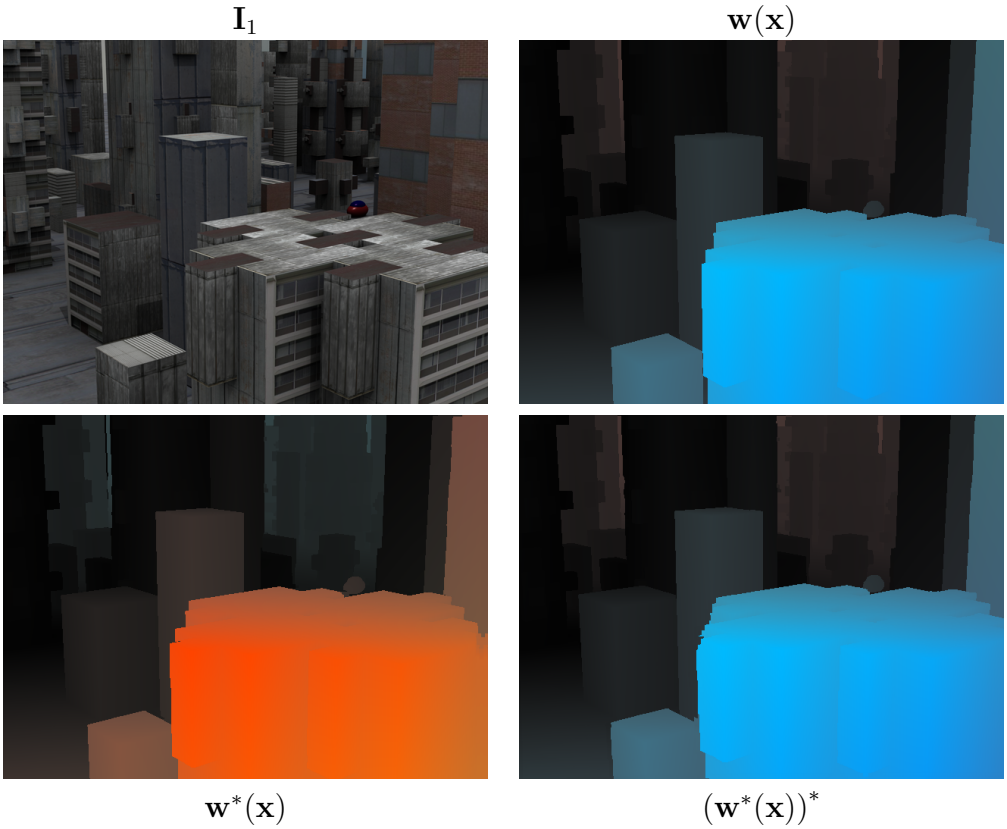


Figure I-8: Reprojection error: first image in the top row, the Urban2 sequence from the Middlebury database; second image, the true flow; first image in the bottom row, the inverse flow, $\mathbf{w}^*(\mathbf{x})$; and, second in the bottom, the inverse of the inverse flow, $(\mathbf{w}^*(\mathbf{x}))^*$.

its simplicity: it only depends on the optical flow, does not use any additional buffer and requires less calculations. The third algorithm has the same features as the first one, but it computes averages with several values per position. Its accuracy is slightly better for most sequences and much better in sequences where the flow is continuous or divergent, like in Dimetrodon and Yosemite.

The best EPE results are obtained for Algorithms 2 and 4. These two rely on the image intensities in order to select the appropriate inverse value. This means that the pixel information is more discriminant and reliable than the flow data.

It is interesting to note that, while the second algorithm outstands in most of the EPE results, the fourth wins in the AAE comparisons. This is because Algorithm 4 is based on the average value, which may benefit a better estimation of the flow direction. This effect is more noticeable in the Dimetrodon and Yosemite sequences as well.

We may conclude that image-based algorithms (Algorithms 2 and 4) outperform algorithms based on the flow (Algorithms 1 and 2). On the other hand, these algorithms

Table I-1: Reprojection error (EPE) for the Middlebury sequences.

Sequence	**Algorithm 1**	**Algorithm 2**	**Algorithm 3**	**Algorithm 4**
Dimetrodon	0,014	*0,006*	0,007	**0,006**
Grove2	0,020	**0,008**	0,018	*0,009*
Grove3	0,094	**0,044**	0,089	*0,045*
Hydrangea	0,019	**0,009**	0,017	*0,010*
RubberWhale	0,010	**0,003**	0,006	*0,004*
Urban2	0,027	*0,011*	0,025	**0,011**
Urban3	0,030	*0,010*	0,027	**0,010**
Venus	0,015	**0,006**	0,015	*0,006*
Yosemite	0,009	0,004	*0,004*	**0,004**

Table I-2: Reprojection error (AAE) for the Middlebury sequences.

Sequence	**Algorithm 1**	**Algorithm 2**	**Algorithm 3**	**Algorithm 4**
Dimetrodon	0,359^o	0,203^o	*0,154^o*	**0,138^o**
Grove2	0,438^o	**0,229^o**	0,466^o	*0,242^o*
Grove3	1,205^o	*0,736^o*	1,233^o	**0,671^o**
Hydrangea	0,463^o	*0,259^o*	0,356^o	**0,250^o**
RubberWhale	0,441^o	*0,195^o*	0,273^o	**0,169^o**
Urban2	0,318^o	*0,163^o*	0,307^o	**0,150^o**
Urban3	0,320^o	**0,171^o**	0,366^o	*0,178^o*
Venus	0,257^o	**0,087^o**	0,257^o	*0,093^o*
Yosemite	0,261^o	0,132^o	*0,122^o*	**0,111^o**

are more complex and require more memory resources than their corresponding flow versions.

The interpolation Algorithms 3 and 4 behave similar to their corresponding nearest neighbor algorithms, but they outperform the latter in the presence of divergent or continuous flow fields, like in Dimetrodon or Yosemite.

figure I-11 compares the solution of a flow-based algorithm (Algorithm 1) with respect to an image-based algorithm (Algorithm 2). In the latter case, we note that some values may fail due to the sampling of the image, as can be seen in the last image. It appears some flow errors in the building because the intensities of the building get confused with the intensities in the background. This may be worse in natural sequences, where the presence of noise and brightness changes is more important. However, this problem only appears in the area of occlusions.

I.1.9 Street-lamp Occlusions

Next, we study the performance of the algorithms with respect to the street-lamp occlusion. We modify two Middlebury sequences, Grove2 and Urban2, adding five static bars in front of the scene. This easily simulates the effect of the objects moving behind a fence.

In figure I-12 we show the results for Grove2. The motion of the objects in this sequence is not big, so the effect of the bars is limited. Nevertheless, we appreciate the width of bars being smaller in the first result (Algorithm 1) than in the second solution (Algorithm 2). This means that the first algorithm has partially selected the motion of the background instead of the motion corresponding to the bars.

Table I-3 presents the EPE and AAE errors for each algorithm. We observe that image-based algorithms clearly outperform flow-based ones. In fact, flow-based methods are unwilling to deal with this kind of occlusions.

Table I-3: Reprojection errors for Grove2 with bars.

Error	**Algorithm 1**	**Algorithm 2**	**Algorithm 3**	**Algorithm 4**
EPE	0,035	**0,008**	0,033	*0,009*
AAE	0,947	**0,201**	1,048	*0,216*

figure I-13 shows the results for the Urban2 sequence with bars. The motion in this sequence is bigger; therefore the effect on the bars is more important. The first algorithm is unable to detect the bars in the front building and they completely disappear in the backward flow. The second algorithm correctly handles the motion of the bars and the rest of objects.

Looking at Table I-4, we observe that the EPE and AAE of image-based algorithms are much smaller than those of flow-based methods. Even, the fourth algorithm provides an important improvement in AAE with respect to the algorithm 2.

Table I-4: Reprojection errors for Urban2 with bars.

Urban2 (5 bars)	**Algorithm 1**	**Algorithm 2**	**Algorithm 3**	**Algorithm 4**
EPE	0,065	*0,011*	0,063	**0,011**
AAE	0,824	*0,163*	0,877	**0,151**

These experiments show that image-based algorithms correctly deal with any kind of occlusions. This is at the expense of using intensity information and an additional buffer for storing the comparisons between the image intensities.

I.1.10 Analysis of Filling Strategies

In this section, we study the behavior of the filling strategies explained in Section I.1.3. The experiments in this section were performed using Algorithm 2.

Tables I-5 and I-6 show the EPE and AAE for the Middlebury sequences, respectively. We have only used the sequences where the flow is also defined at occlusions. If we compare these results with the results presented in Tables I-1 and I-2, we observe that the errors increased considerably.

Table I-5: Reprojection errors (EPE) for the filling strategies.

Sequence	**Minimum fill**	**Average fill**	**Oriented fill**
Grove2	**0,023**	0,039	0,040
Grove3	**0,166**	0,264	0,206
Urban2	0,083	0,130	**0,041**
Urban3	0,149	0,166	**0,055**
Venus	0,021	0,038	**0,017**
Yosemite	0,008	0,006	**0,005**
Grove2 with bars	0,092	**0,065**	0,094
Urban2 with bars	0,160	**0,132**	0,160

Looking at these tables, we conclude that the best performing strategies are the oriented-fill and the min-fill approaches. The oriented-fill approach seems to have a better behavior. The average-fill strategy behaves better for the sequences with the static bars (two last sequences).

Table I-6: Reprojection errors (AAE) for the filling strategies.

Sequence	**Minimum fill**	**Average fill**	**Oriented fill**
Grove2	**0,595**^{o}	1,010^o	1,014^o
Grove3	**1,972**^{o}	3,417^o	2,761^o
Urban2	0,472^o	1,293^o	**0,371**^{o}
Urban3	1,725^o	2,262^o	**0,676**^{o}
Venus	0,325^o	0,791^o	**0,284**^{o}
Yosemite	0,171^o	0,184^o	**0,133**^{o}
Grove2 with bars	3,062^o	**1,346**^{o}	3,112^o
Urban2 with bars	2,851^o	**1,129**^{o}	2,878^o

This means that this strategy is better for the street-lamp occlusions. This kind of occlusions is difficult to deal with, because they produce a disocclusion at a place that

may be far from the object that originated it (see figure I-13 for example). The minimum and the oriented strategies may not assure a correct filling.

In any case, dealing with the filling of the street-lamp case is not easy. An alternative would be to use a maximum-fill strategy, but there is no way to differentiate between the stereoscopic and the street-lamp occlusions. A good approach would be to use the oriented-fill for the stereoscopic and the average-fill for the street-lamp case, but we have not found an easy mechanism to differentiate between these two situations.

I.1.11 Recursive Application

In this section, we study the effect of applying the algorithms multiple times. This will give us an idea of how the solutions degrade when we apply the methods recursively. For this experiment, we use the Yosemite and the Urban3 sequences.

In Tables I-7 and I-8 we show the EPE and AAE results for the Yosemite sequence, respectively. We compare the solutions applying the inverse optical flow 2 times ($\mathbf{w}^{2*}(\mathbf{x})$), 10 times ($\mathbf{w}^{10*}(\mathbf{x})$), 20 times ($\mathbf{w}^{20*}(\mathbf{x})$) and 100 times ($\mathbf{w}^{100*}(\mathbf{x})$). In this case, we use the average-fill strategy.

Table I-7: Reprojection errors (EPE) for the Yosemite sequence with recursive application.

Error	**Algorithm 1**	**Algorithm 2**	**Algorithm 3**	**Algorithm 4**
$\mathbf{w}^{2*}(\mathbf{x})$	0,011	0,006	0,006	**0,006**
$\mathbf{w}^{10*}(\mathbf{x})$	0,046	**0,011**	0,018	0,011
$\mathbf{w}^{20*}(\mathbf{x})$	0,094	**0,012**	0,037	0,014
$\mathbf{w}^{100*}(\mathbf{x})$	0,614	**0,013**	0,395	0,026

The results for Algorithms 2 and 4 are surprisingly very stable. In fact, the former hardly degrades the results and, even computing the inverse flow 100 times, we obtain a very similar result. On the contrary, Algorithms 1 and 3 strongly degrade the solutions, the first one being the worst.

Table I-8: Reprojection errors (AAE) for the Yosemite sequence with recursive application.

Error	**Algorithm 1**	**Algorithm 2**	**Algorithm 3**	**Algorithm 4**
$\mathbf{w}^{2*}(\mathbf{x})$	$0{,}321^o$	$0{,}182^o$	$0{,}184^o$	$\mathbf{0{,}155}^o$
$\mathbf{w}^{10*}(\mathbf{x})$	$1{,}141^o$	$0{,}287^o$	$0{,}458^o$	$\mathbf{0{,}265}^o$
$\mathbf{w}^{20*}(\mathbf{x})$	$2{,}163^o$	$\mathbf{0{,}316}^o$	$0{,}794^o$	$0{,}325^o$
$\mathbf{w}^{100*}(\mathbf{x})$	$12{,}101^o$	$\mathbf{0{,}335}^o$	$7{,}350^o$	$0{,}620^o$

figure I-14 shows the results $\mathbf{w}^{10*}(\mathbf{x})$, $\mathbf{w}^{20*}(\mathbf{x})$ and $\mathbf{w}^{100*}(\mathbf{x})$ for Yosemite. In the first row, we show the results for Algorithm 1 and, in the second, the results for Algorithm 2. The last row shows that the latter image-based algorithm is stable, while the first shows the unstability of flow-based algorithms.

Similar results can be seen in Tables I-9 and I-10 for the Urban3 sequence. In this case we have used the oriented-fill strategy.

Table I-9: Reprojection errors (EPE) for the Urban3 sequence with recursive application.

Error	**Algorithm 1**	**Algorithm 2**	**Algorithm 3**	**Algorithm 4**
$\mathbf{w}^{2*}(\mathbf{x})$	0,058	0,053	0,055	**0,052**
$\mathbf{w}^{10*}(\mathbf{x})$	0,225	0,082	0,211	**0,082**
$\mathbf{w}^{20*}(\mathbf{x})$	0,438	**0,093**	0,409	0,093
$\mathbf{w}^{100*}(\mathbf{x})$	1,957	**0,100**	1,907	0,104

Table I-10: Reprojection errors (AAE) for the Urban3 sequence with recursive application.

Urban3 (AAE)	**Algorithm 1**	**Algorithm 2**	**Algorithm 3**	**Algorithm 4**
$\mathbf{w}^{2*}(\mathbf{x})$	$\mathbf{0,575}^o$	$0,856^o$	$0,676^o$	$0,895^o$
$\mathbf{w}^{10*}(\mathbf{x})$	$1,973^o$	$\mathbf{1,351}^o$	$2,320^o$	$1,354^o$
$\mathbf{w}^{20*}(\mathbf{x})$	$3,750^o$	$1,551^o$	$4,288^o$	$\mathbf{1,526}^o$
$\mathbf{w}^{100*}(\mathbf{x})$	$12,026^o$	$1,693^o$	$12,446^o$	$\mathbf{1,600}^o$

We can see in figure I-15 that the results of Algorithm 2 (second row) degrades less severely than the results of Algorithm 1 (first row). These results seem reasonable, since flow-based algorithms use the flow information as discriminant. This information degrades with every inverse calculation, so the errors accumulate in every iteration. On the other hand, image-based algorithms rely on the image information, which is not modified.

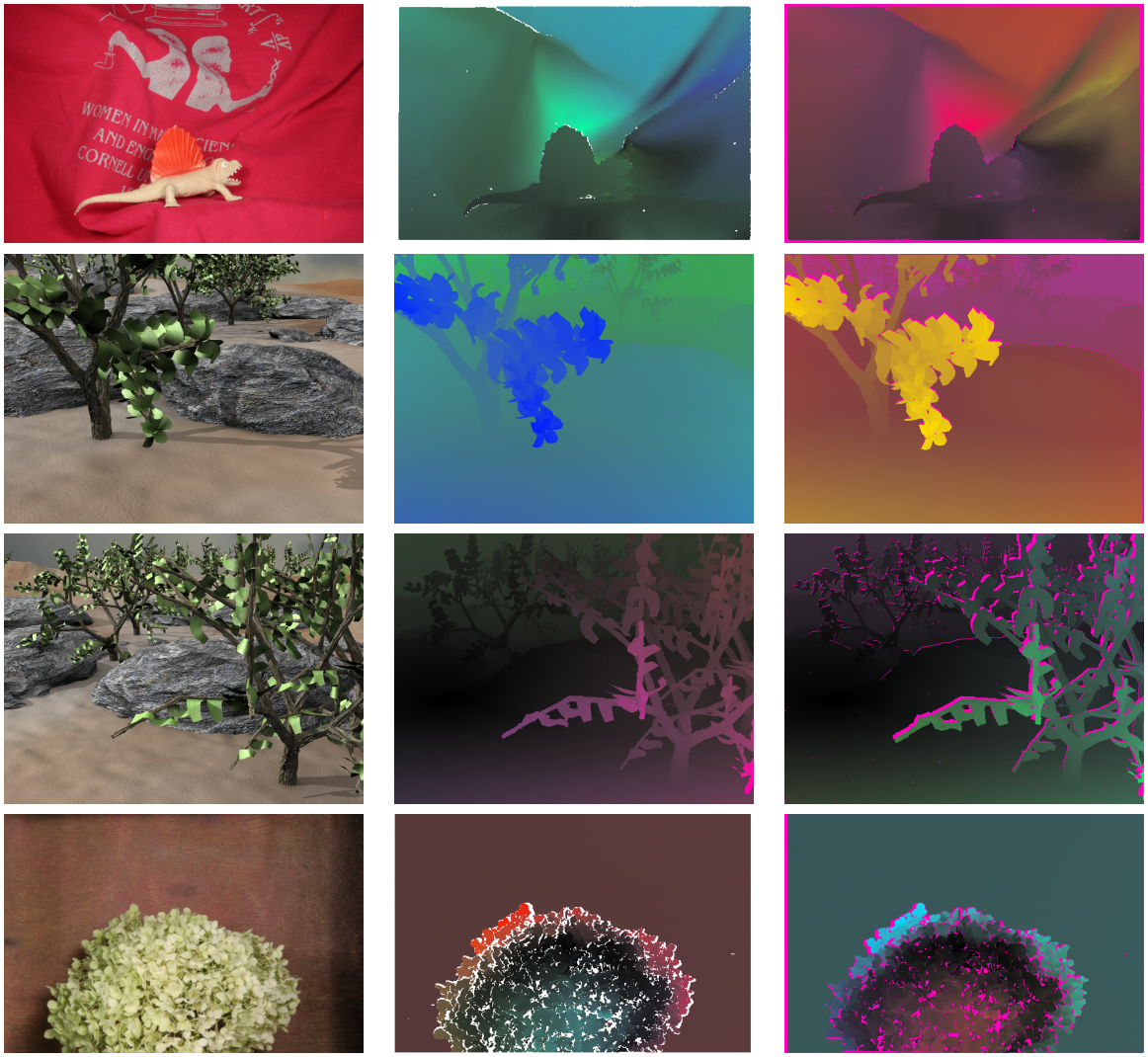


Figure I-9: Middlebury test sequences. First column, the source image; second, the true flow; third, the inverse optical flow using Algorithm 1, with disocclusions in pink.

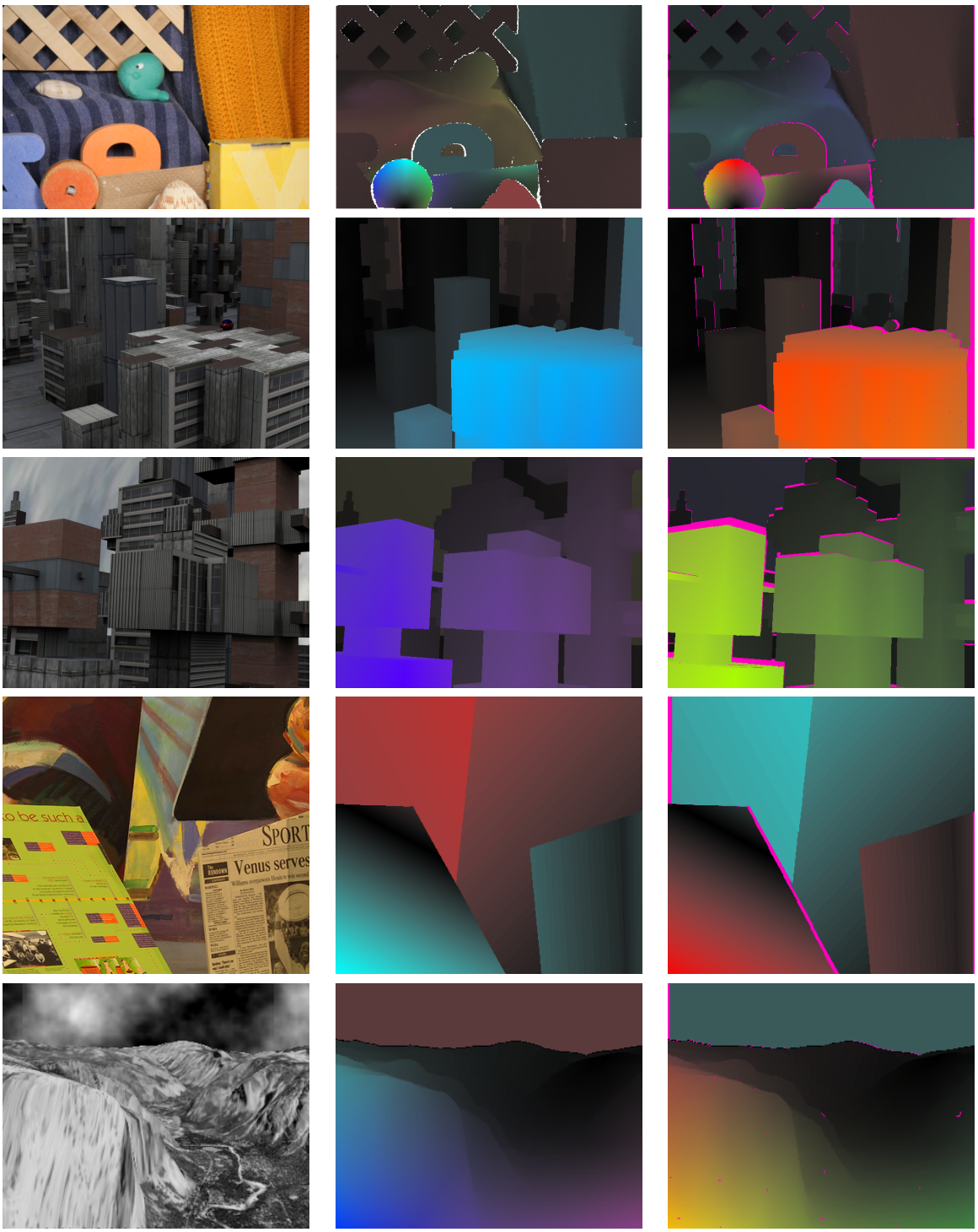


Figure I-10: Middlebury test sequences. First column, the source image; second, the true flow; third, the inverse optical flow using Algorithm 1, with disocclusions in pink.

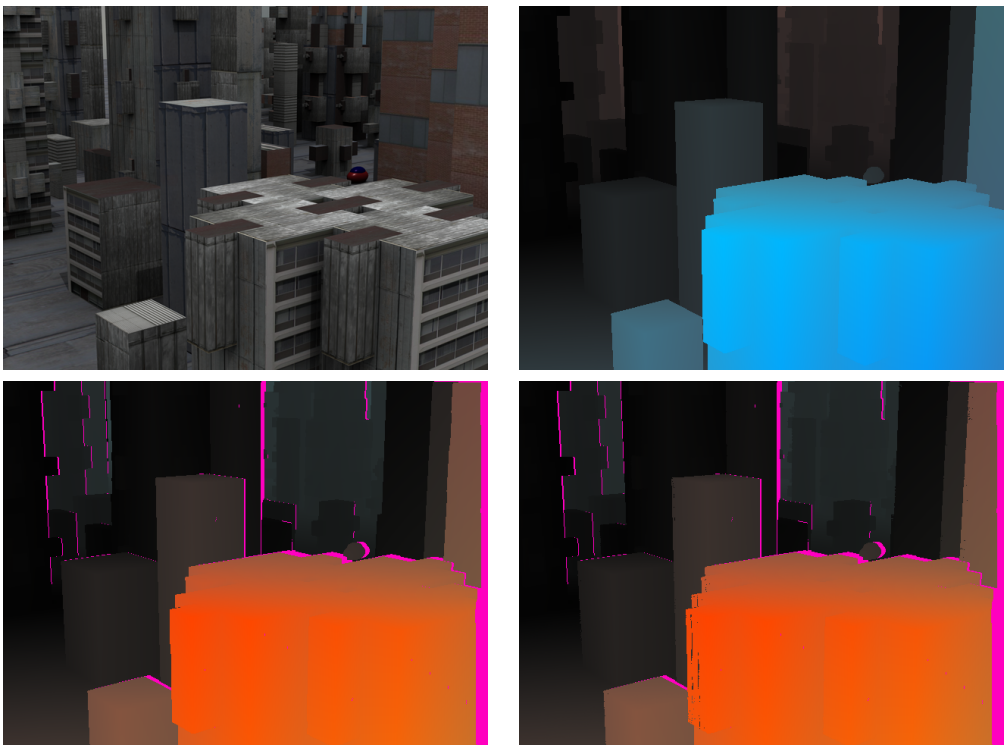


Figure I-11: Flow-based versus image-based algorithms. Comparison between Algorithms 1 and 2.

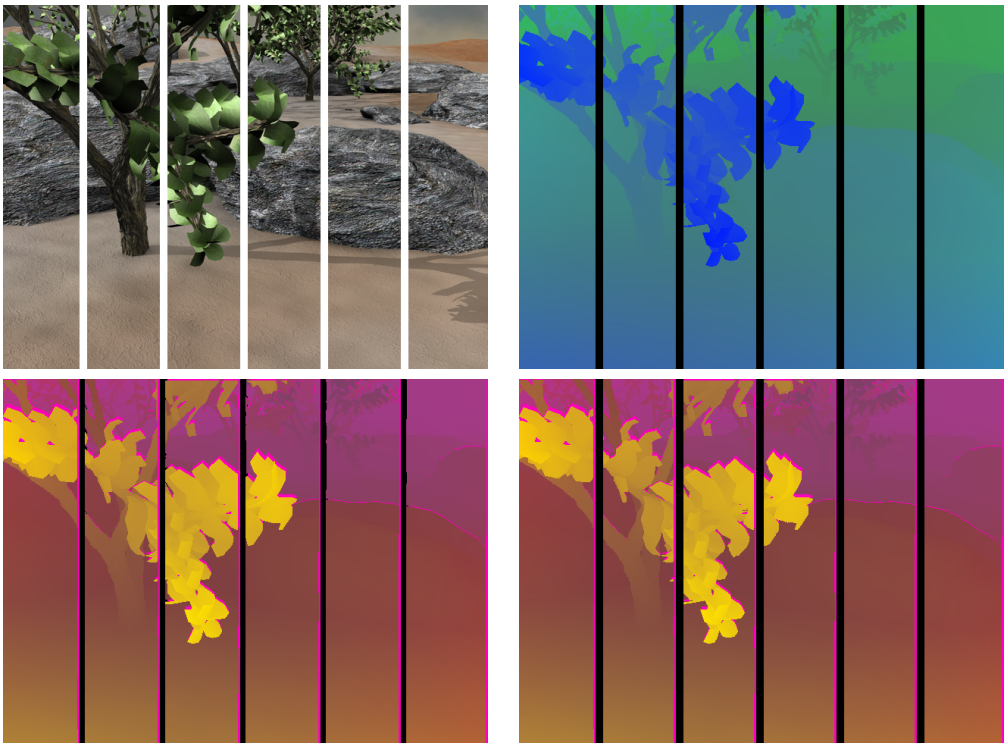


Figure I-12: Street-lamp occlusion for Grove2. First row, the source image and the true flow. Second row, the backward flows for Algorithms 1 and 2.

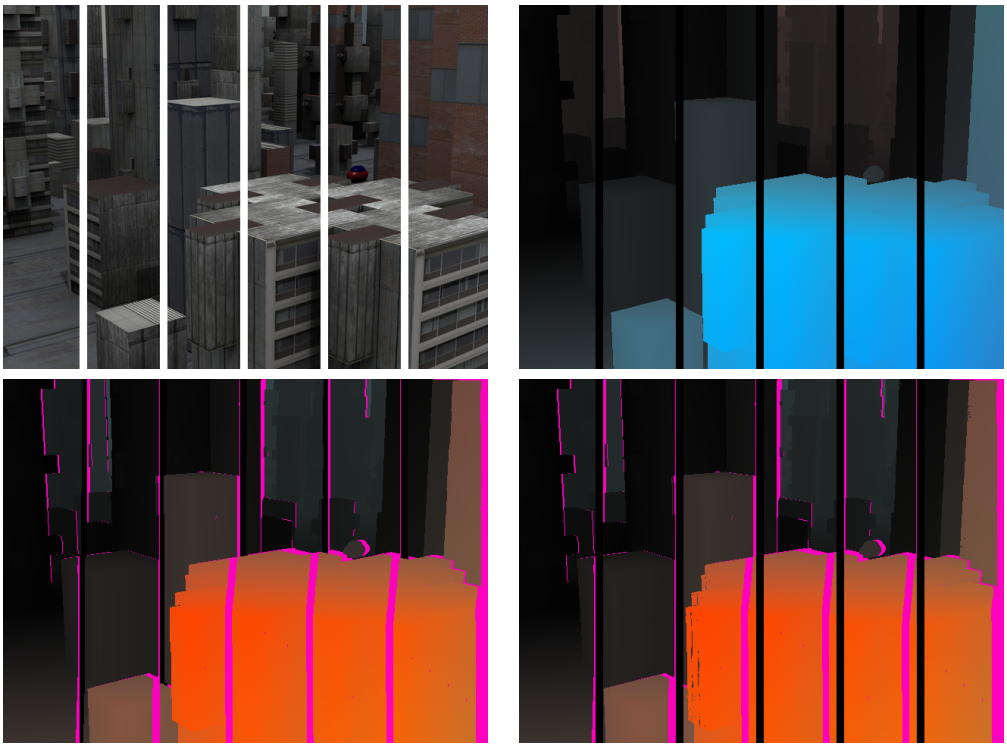


Figure I-13: Street-lamp occlusion for Urban2. First row, the source image and the true flow. Second row, the backward flows for Algorithms 1 and 2.

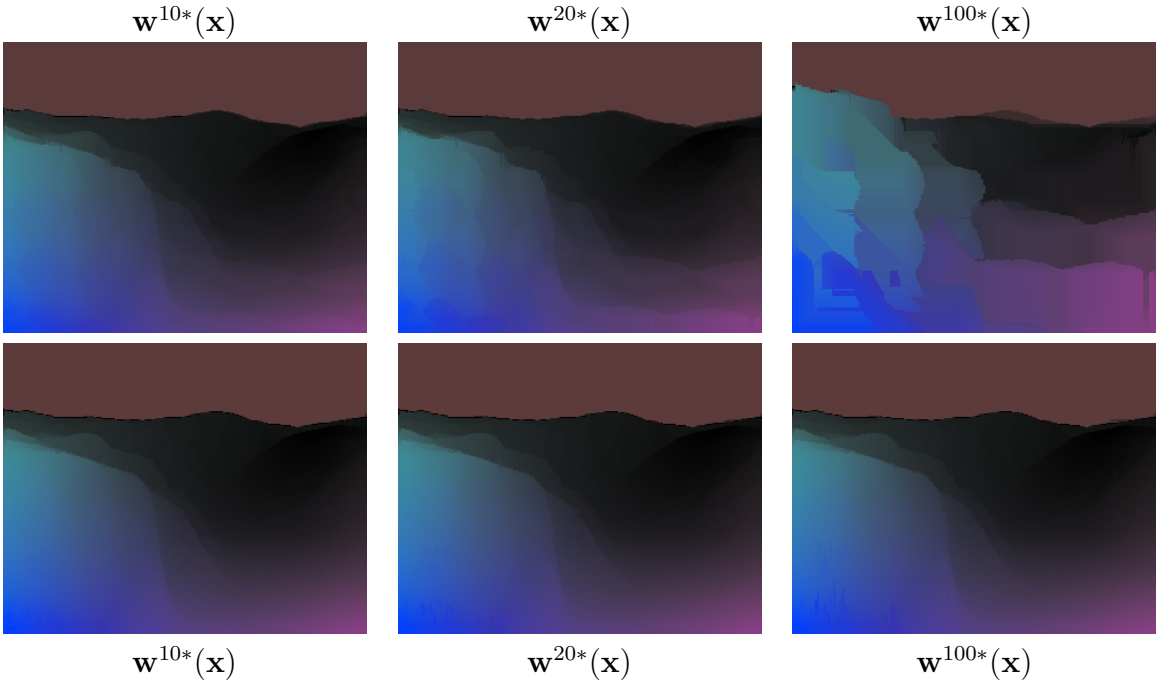


Figure I-14: Yosemite recursive. First row, results for Algorithm 1. Second row, results for Algorithm 2.

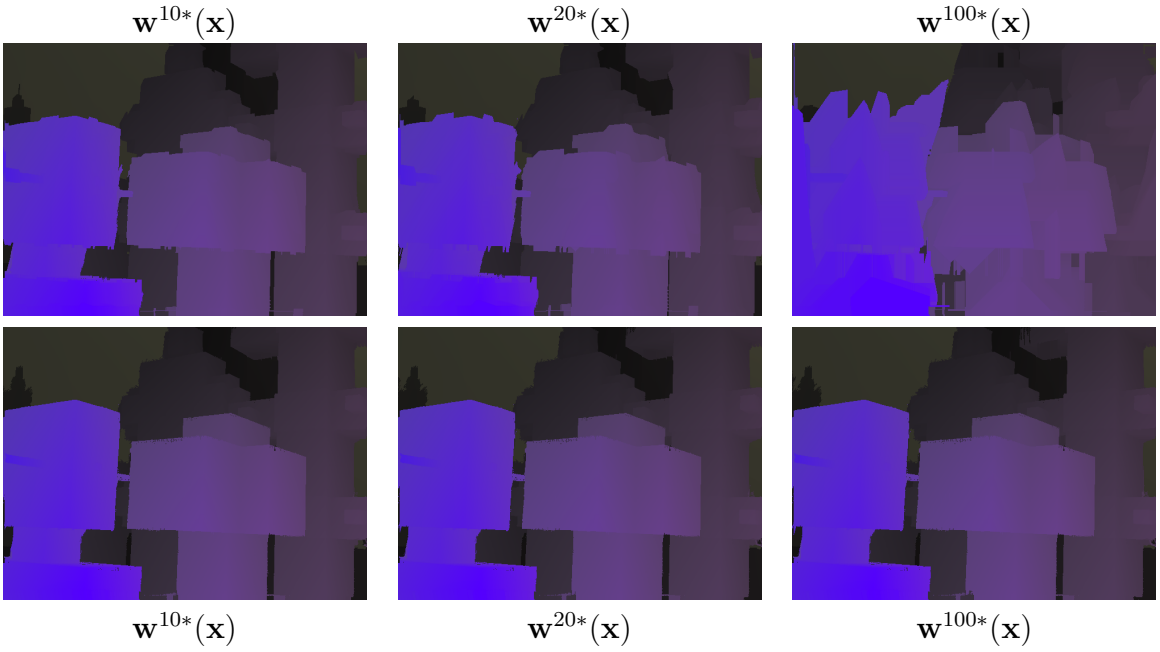


Figure I-15: Urban3 recursive. First row, results for Algorithm 1. Second row, results for Algorithm 2.

I.2 Summary and Conclusions

We have proposed four new efficient algorithms for estimating the inverse optical flow. We have classified these in flow-based methods, that only use the forward optical flow as input, and image-based methods, that also use the information of the images. The former are limited for sequences with stereoscopic occlusions, whereas the latter are more general and can also deal with the street-lamp occlusion case.

We have seen the importance of occlusions and disocclusions in the estimation of the backward flow. All the algorithms can easily handle occlusions, but disocclusions have to be processed separately. In order to fill dissoclusions, we have proposed three filling strategies, based on the minimum and average flows, and an oriented filling process.

Flow-based algorithms are easy to implement and efficient in running time and memory requirements. However, image-based algorithms yield the most accurate results, at the expense of using more buffers and calculations. Typically, we need a buffer for dealing with occlusions. On the other hand, nearest neighbor algorithms need one image pass, while interpolation algorithms need two image passes. The latter also need more buffers to store the accumulative values.

We have shown that the accuracy of the methods is high in general. If we do not take into account disocclusions, then the errors are smaller than 1% of a pixel in Euclidean distance and less than 1^o in angular error. Errors increase considerably when we deal with disocclusions. The oriented fill strategy seems to provide the best results, followed by the min-fill strategy. Nevertheless, in the presence of street-lamp occlusions, the average fill strategy outperforms the other approaches.

We have also shown that image-based algorithms are also more stable. When we calculate the inverse flow many times, the results of flow-based methods rapidly degrade. This is due to the fact that the flow itself is modified and the errors are accumulated. This kind of methods might only be used when a few iterations are needed. These are interesting if speed or memory requirements are the first considerations.

Although the proposed algorithms are efficient, they can still be improved. The major effort should be oriented to improve the filling strategies. This problem can be tackled using more temporal information, so that the unknown information can be discovered from a different time instant. Another improvement is to integrate the filling process with the main part of the algorithm, as it happens with occlusions.

Appendix II: Details of implementation of the ROF method

In this appendix we present pseudo-codes and a brief explanation of the parameters of our multichannel implementation of the IPOL-Brox method[1] [Sánchez13b].

The algorithm implements the numerical scheme in equation (4.10) for a multi-channel framework. It takes a set of color images as input data and computes the optical flows between every pair of consecutive images. We separate the algorithm in two modules: one procedure that computes the optical flows in each scale, and the main algorithm that is in charge of handling the pyramidal structure. This method depends on the parameters given in table II-1.

In the procedure, `MAXITER` is the maximum number of iterations allowed for the convergence of the SOR method. Its value is constant and is high enough to let the method converge. In the source code, the SOR loop is unrolled in order to avoid boundary tests when computing $div(du^{m+1})$ and $div(dv^{m+1})$. This means that the first and last columns and rows, and the four corners of the images, are computed separately.

In theory, as we have seen before, the method needs three levels of iterations. Nevertheless, we have found in the experiments that the inner iterations can be integrated in the outer iterations without a loss of precision. For this reason we have decided to estimate the values of Ψ'_S, div_u, div_v and div_d inside the outer iterations. We prefer this option in order to avoid the estimation of ∇du. The other computations, at the beginning of the inner loop, may also be integrated in the outer loop.

The main process is given in algorithm 6. In order to turn the method more stable to the input parameters, it first normalizes the images between 0 and 255; it convolves the finest scale images with a small Gaussian kernel; then, it creates the pyramidal structure for the whole sequence; and, finally, it goes over the set of scales computing the optical flows at different resolutions.

[1] http://www.ipol.im/pub/art/2013/21/

Table II-1: Parameters of the method

Parameter	**Explanation**
α	Regularization parameter. It determines the smoothness of the output. The bigger this parameter is, the smoother the solutions we obtain.
γ	Parameter that determines the influence of the gradient constancy assumption.
N_{scales}	Number of scales in the pyramidal structure. If the flow field is very small (about one pixel), it can be set to 1. Otherwise, it should be set so that $(1/\eta)^{N-1}$ is larger than the expected size of the largest displacement.
η	Downsampling factor. It is used to downscale the original images in order to create the pyramidal structure. Its value must be in the interval $(0, 1)$. With $\eta = 0.5$, the images are reduced to half their size in each dimension from one scale to the following.
ε	Stopping criterion threshold. It is the threshold used to stop the SOR iterations, given in equation (4.11).
inner_iterations	Number of inner iterations in the numerical scheme. It corresponds to index n in equation (4.10).
outer_iterations	Number of outer iterations in the numerical scheme. It corresponds to index m in equation (4.10).

Procedure BroxOpticFlow($I_1^c, I_2^c, u, v, \alpha, \gamma, \varepsilon, inner\_iterations, outer\_iterations$)

Input: $I_1^c, I_2^c, \alpha, \gamma, \varepsilon := 0.001, inner\_iterations := 1, outer\_iterations := 10, \omega := 0.9$

Output: u, v

Compute, for all channels, $I_{1,x}^c, I_{1,y}^c, I_{2,x}^c, I_{2,y}^c$
Compute, for all channels, $I_{2,xx}^c, I_{2,yy}^c, I_{2,xy}^c$
for $no \leftarrow 0$ **to** $outer\_iterations - 1$ **do**
    Compute $I_2^c(\mathbf{x}+\mathbf{w}), I_{2,x}^c(\mathbf{x}+\mathbf{w}), I_{2,y}(\mathbf{x}+\mathbf{w})$ using bicubic interpolation
    Compute $I_{2,xx}^c(\mathbf{x}+\mathbf{w}), I_{2,xy}^c(\mathbf{x}+\mathbf{w}), I_{2,yy}^c(\mathbf{x}+\mathbf{w})$ using bicubic interpolation
    Compute u_x, u_y, v_x, v_y
    Compute Ψ'_S using equation (4.3)
    Compute div_u, div_v, div_d using equations (4.6) and (4.8)
    $du \leftarrow 0$
    $dv \leftarrow 0$
    for $ni \leftarrow 0$ **to** $inner\_iterations - 1$ **do**
        Compute Ψ'_D, Ψ'_G using equation (4.3)
        Compute Au, Av, Du, Dv, D using equation (4.9)
        while $error > \varepsilon$ **and** $nsor < MAXITER$ **do**
            $du \leftarrow (1-\omega)\, du + \omega\, (Au - D\, dv + \alpha\, div\_du)\, /Du$
            $dv \leftarrow (1-\omega)\, dv + \omega\, (Av - D\, du + \alpha\, div\_dv)\, /Dv$
            Compute $error$ with equation (4.11)
            $nsor \leftarrow nsor + 1$
        end
    end
    $u \leftarrow u + du$
    $v \leftarrow v + dv$
end

Algorithm 5: Pyramidal structure management

Input: $I_1^c, I_2^c, u, v, \alpha, \gamma, N_{scales}, \eta, \varepsilon, inner\_iterations, outer\_iterations$

Output: u, v

Normalize multi-channel images between 0 and 255

Convolve the images with a Gaussian of $\sigma \leftarrow 0.8$

Create the pyramid of images $I_1^{c,s}, I_2^{c,s}$ using η (with $s \leftarrow 0, \ldots, N_{scales} - 1$)

$\alpha_c \leftarrow \alpha \cdot nchannels$

for $s \leftarrow N_{scales} - 1$ **to** 0 **do**

  `BroxOpticFlow`$(I_1, I_2 u^s, v^s, \alpha, \gamma, inner\_iterations, outer\_iterations)$

  if $s > 0$ **then**

    $u^{s-1}(\mathbf{x}) := \frac{1}{\eta} u^s(\eta \mathbf{x})$

    $v^{s-1}(\mathbf{x}) := \frac{1}{\eta} v^s(\eta \mathbf{x})$

  end

end

Appendix III: Details of implementation of the RDPOF method

Here, we present several details and the parameters of the discontinuity preserving strategies proposed in this dissertation. Table III-1 shows the three alternatives: the first corresponds to the exponential approximation while the others are the two proposals for solving the instability problems. We named the strategies *DF*, *DF-β* and *DF-Auto*, respectively. The regularization behavior relies on $\Psi(\nabla I_1)$; thus, we show the difference in the second column of the table.

Strategy	$\Psi(\nabla I_1)$	Index
DF	$e^{-\lambda\|\nabla I_1\|}$	1
DF-β	$e^{-\lambda\|\nabla I_1\|} + \beta$	2
DF-Auto	$e^{-\lambda_{auto}\|\nabla I_1\|}$	3

Table III-1: Regularization strategies. The first alternative offers regularization with a decreasing scalar function. The second ensures constant diffusion when using the previous scheme. The third provides an automatic adaptation of the parameter that controls the decreasing scalar function.

We describe the algorithm that implements the numerical scheme in Equation (4.10). It takes two color images as input data and computes the optical flow. Our algorithm depends on the parameters given in Table III-2 and the constants of Table III-3. The first parameter is an integer for selecting the diffusion behavior (see Table III-1). The weighting parameters used in the energy functional are α and γ; λ is used in the non-automatic strategies for controlling the diffusion at image borders.

The remaining parameters are the following: N_{scales}, η stand for the number of scales and the downsampling factor in the pyramidal scheme and σ_0 used in the image convolution; the parameters for the numerical scheme, given by the inner and outer

iterations, and the stopping criterion threshold (ε); β, ξ and τ are constants that support the regularization strategies as explained in the previous section.

Table III-2: Parameters of the method.

Parameter	**Explanation**
method_type	Integer that selects the regularization strategy (see Table III-1).
α	Regularization parameter. It determines the smoothness strength. The bigger this parameter is, the smoother the solutions we obtain. In our algorithm, it is adapted to the number of channels.
γ	Parameter that determines the influence of the gradient constancy assumption.
λ	Used in *DF* and *DF-β* methods. It determines the influence of the exponential function in the regularization. Its value is automatic using the *DF-Auto* approach.

We separate the algorithm in two modules: one procedure that computes the optical flow at each scale and the main algorithm that is in charge of handling the pyramidal structure. The main Algorithm 6 creates the pyramidal structure, in a very similar way to the ROF method, and calls Algorithm 7 for computing the optical flow, starting from the coarsest scale. It adapts the result at each scale to be used as the initial approximation at the following scale.

Algorithm 7 calculates Ψ using the regularization scheme defined by the parameter *method_type*. The result is used before the inner-iterations to calculate the Ψ equations using procedure `ExponentialCalculation`.

When using *DF* or *DF-β* strategies, the program looks for the maximum gradient in every pixel of the multichannel image in order to calculate the exponential function.

On the other hand, for the *DF-Auto* approach, we use procedure `automatic_lambda` to calculate λ and the maximum gradient per pixel for the exponential, as in the previous case. This permits to accelerate the method by doing a unique search over the image. This allows us to solve both problems at once. Finally, we have used the OpenMP library to parallelize the algorithm making it faster and achieving fast running times (see the on-line demo[2]).

[2]http://demo.ipol.im/demo/172/

Table III-3: Constant parameters of the method.

Parameter	**Explanation**
N_{scales}	Number of scales in the pyramidal structure. If the flow field is very small (about one pixel), it can be set to 1. Otherwise, it should be set so that $(1/\eta)^{N-1}$ is larger than the expected size of the largest displacement. N_{scales} is automatically calculated so that the image size, at the coarsest scale, is around 16×16 pixels.
η	Downsampling factor. It is used to downscale the original images in order to create the pyramidal structure. Its value must be in the interval $(0, 1)$. With $\eta = 0.5$, the images are reduced to half their size in each dimension from one scale to the following. We fix it to $\eta := 0.75$.
σ_0	It is used in the image convolution made before the optical flow calculation. We use $\sigma_0 := 0.6$.
ε	Stopping criterion threshold. It is the threshold used to stop the SOR iterations, given in Equation (4.11). Its value is $\varepsilon := 0.001$.
inner_iter	Number of inner iterations in the numerical scheme. It corresponds to index n in Equation (4.10 used in this method). We use *inner_iter* $:= 1$
outer_iter	Number of outer iterations in the numerical scheme. It corresponds to index m in Equation (4.10 used in this method). We use *outer_iter* $:= 10$
β	Used in *DF-*β method. It is a constant that ensures a minimum diffusion. We fix its value to $\beta := 0.001$.
ξ	Used in *DF-Auto* method. It is a constant that determines if the diffusivity is sufficient to avoid the ill-posedness. Its value is $\xi := 0.05$.
τ	Used in *DF-Auto* method. It is a constant that determines the conservative behavior of the automatic λ. It prevents the occurrence of instabilities. We fix it value to $\tau := 0.94$.

Algorithm 6: Pyramidal structure management

Input: $I_1^c, I_2^c, u, v, method\_type, \alpha, \gamma, \lambda, N_{scales}, \eta = 0.5, \varepsilon := 0.001, inner\_iter := 1, outer\_iter := 10, nchannels$

Output: u, v

Normalize multi-channel images between 0 and 255
Convolve the images with a Gaussian of $\sigma \leftarrow 0.8$
Create the pyramid of images $I_1^{c,s}, I_2^{c,s}$ using η (with $s \leftarrow 0, \ldots, N_{scales} - 1$)
$\alpha_c \leftarrow \alpha \cdot nchannels$
for $s \leftarrow (N_{scales} - 1)$ **to** 0 **do**
    `RobustDFmethods(`$I_1^{c,s}, I_2^{c,s}, u^s, v^s, method\_type, \alpha_c, \gamma, \lambda, inner\_iter, outer\_iter$`)`
    if $s > 0$ **then**
        $u^{s-1}(\mathbf{x}) \leftarrow \frac{1}{\eta}\, u^s(\eta\, \mathbf{x})$
        $v^{s-1}(\mathbf{x}) \leftarrow \frac{1}{\eta}\, v^s(\eta\, \mathbf{x})$
    end
end

Algorithm 7: RobustDFmethods($I_1^c, I_2^c, u, v, method\_type, \alpha, \gamma, \lambda, inner\_iter, outer\_iter$)

Input: $I_1^c, I_2^c, method\_type, \alpha, \gamma, \lambda, \varepsilon := 0.001, inner\_iter := 1, outer\_iter := 10, \omega := 0.9$

Output: u, v

Compute, for all channels, $I_{1,x}^c, I_{1,y}^c, I_{2,x}^c, I_{2,y}^c$
Compute, for all channels, $I_{2,xx}^c, I_{2,yy}^c, I_{2,xy}^c$
`ExponentialCalculation` $(I_{1x}, I_{1y}, \alpha, \lambda, method\_type, \Phi)$
for $no \leftarrow 0$ **to** $outer\_iter - 1$ **do**
    Compute $I_2^c(\mathbf{x}+\mathbf{w}), I_{2,x}^c(\mathbf{x}+\mathbf{w}), I_{2,y}(\mathbf{x}+\mathbf{w})$ using bicubic interpolation
    Compute $I_{2,xx}^c(\mathbf{x}+\mathbf{w}), I_{2,xy}^c(\mathbf{x}+\mathbf{w}), I_{2,yy}^c(\mathbf{x}+\mathbf{w})$ using bicubic interpolation
    Compute the flow gradients u_x, u_y, v_x, v_y
    Compute ψ'_S and Ψ
    Compute div_u, div_v, div_d using equations (4.6) and (4.8)
    $(du, dv) \leftarrow (0, 0)$
    for $ni \leftarrow 0$ **to** $inner\_iter - 1$ **do**
        Compute ψ'_D, ψ'_G
        Compute Au, Av, Du, Dv, D using Equation (4.9)
        $nsor \leftarrow 0$
        while $error > \varepsilon$ **and** $nsor < MAXITER$ **do**
            $du \leftarrow (1-\omega)\, du + \omega \cdot \frac{Au - D\, dv + \alpha\, div\_du}{Du}$
            $dv \leftarrow (1-\omega)\, dv + \omega \cdot \frac{Av - D\, du + \alpha\, div\_dv}{Dv}$
            Compute $error$ with Equation (4.11)
            $nsor \leftarrow nsor + 1$
        end
    end
    $(u, v) \leftarrow (u + du, v + dv)$
end

Procedure ExponentialCalculation($I^c_{1,x}, I^c_{1,y}, \alpha, \lambda, \beta, method\_type, \Phi$)

Input: $I^c_{1,x}, I^c_{1,y}, \alpha, \lambda, \beta := 0.001, method\_type$
Output: Φ

```
switch method_type do
    case 1:2
        β_v ← 0
        if method_type = 2 then β_v ← β
        Compute maximum gradient (g_max)
        foreach pixel p do
            Φ(p) ← e^(−λ·g_max) + β_v
        end
    case 3:
        λ_Ω ← AutomaticLambda (I_x, I_y, α, λ_p, g_max)
        foreach pixel p do
            λ_auto ← λ_Ω
            if λ_Ω > λ_p(p) then
                λ_auto ← λ_p(p)
            end
            Φ(p) ← e^(−λ_auto·g_max)
        end
    end
endsw
```

Procedure AutomaticLambda

Input: $I_x, I_y, \alpha, \tau := 0.94, \xi := 0.05$

Output: λ_Ω, λ_p, g_{max}

foreach *pixel* p **do**

  $g_{max}(p) \leftarrow$ Compute the maximum gradient for each image channel at every pixel

  $\lambda(p) \leftarrow \frac{-\log(\xi)+\log(\alpha)}{g_{max}(p)}$

  if $max\_gradient(p) < g_{max}(p)$ **then** $max\_gradient(p) \leftarrow g_{max}(p)$

end

$gradient\_sorted \leftarrow sort(max\_gradient)$

$pos\_ref \leftarrow \tau \cdot image\_width \cdot image\_height$

$\lambda_\Omega \leftarrow \frac{-\log(\xi)+\log(\alpha)}{gradient\_sorted(pos\_ref)}$

return λ_Ω

Appendix IV: Work flow for the RDPOF IPOL demo

In this section, we explain the work flow and details of the demo associated with the IPOL article: 'Robust Discontinuity Preserving Optical Flow Methods'.

We show in figure IV-1 the main page of the paper. Here, the readers can see typical information such as the authors, date of the publication, number of pages or the abstract. Furthermore, they can also download the research work in different resolutions and the original source code.

Figure IV-2 shows the main page of the corresponding on-line demo. Note that in this web page, the users can select a pair of consecutive images or upload their own ones for the optical flow calculation. Once the users make the selection, the images are preprocessed, their size checked and converted to the image file format expected by the software shown in the demo. Then, the system guides the work flow into another step where the users can set the input parameters and choose the optical flow algorithm that they want to use (figure IV-3).

Next, the input images are processed by the software according to the user decisions. Once the flow calculation is finished, the result is shown as we can see in figure IV-4). In this particular demo, the system shows different results and a summary of information such as running time and errors. This particular sequence does not have true flow so, the demo can only calculate the Average Backprojection Error. Note that the demo allows showing the results with other schemes by changing the values of the object at the top right corner of the figure.

On the other hand, if the input images were submitted by the user, this experiment is archived storing the given images, parameters and data that the authors of the demo want to store. In this sense, you can access this archive at any moment visualizing a web page similar to the one depicted in figure IV-5. In our opinion, the archive functionality is quite interesting because any user can determine if the algorithm fits well for his particular needs. This simple idea extend the dimension of the authors publication. In our particular case, we used standard images in the paper for our experiments. However, the on-line demo allows that any researcher can use our proposals into medical images, stereo datasets

or any other option.

Finally, note that this work flow will be similar for any other IPOL work but adapted for the features of a particular demo.

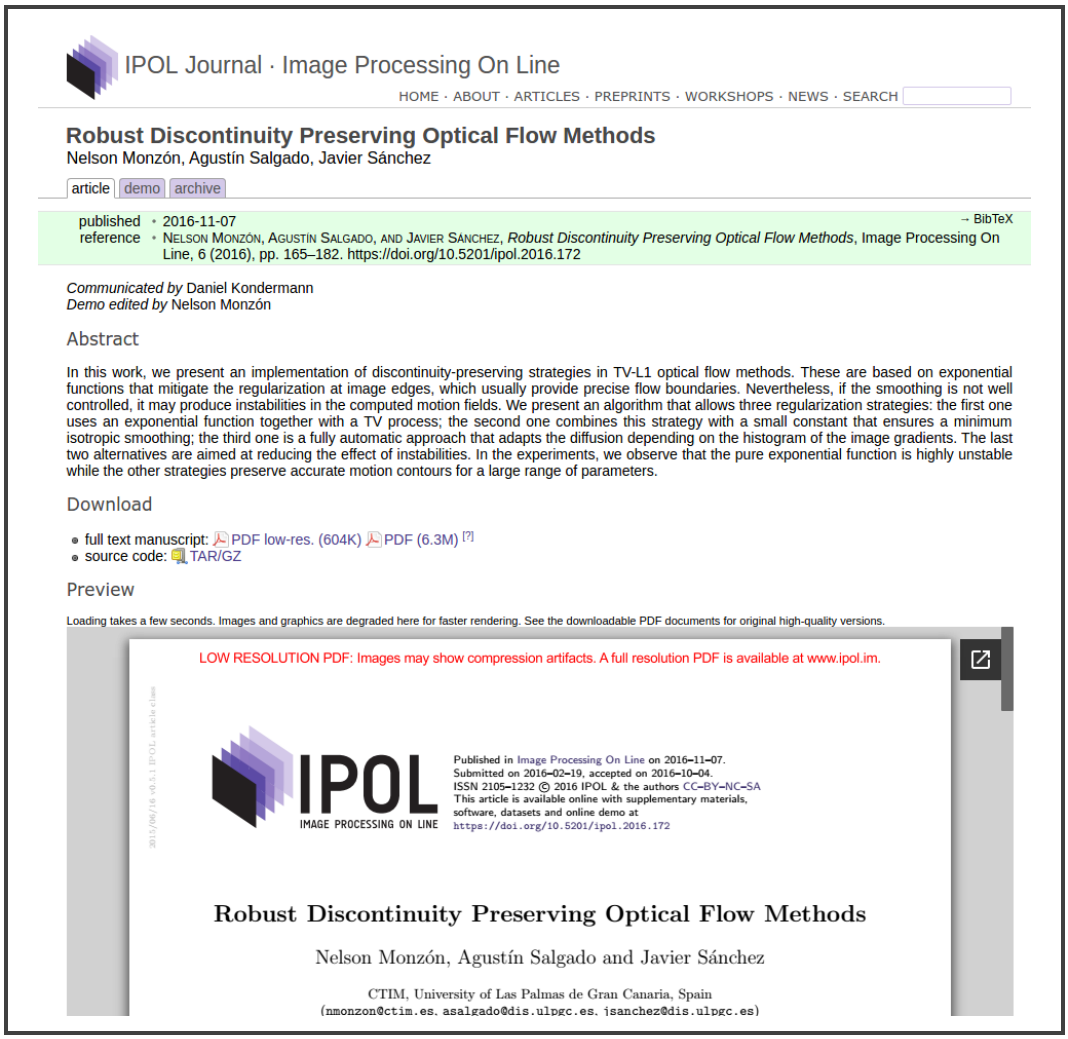


Figure IV-1: Main page of the on-line publication 'Robust Discontinuity Preserving Optical Flow Methods'. The main page depicts the paper to visualize its contents on-line or you can download the paper in full and low-resolution. You can also download the original code that is the one used in the on-line demo.

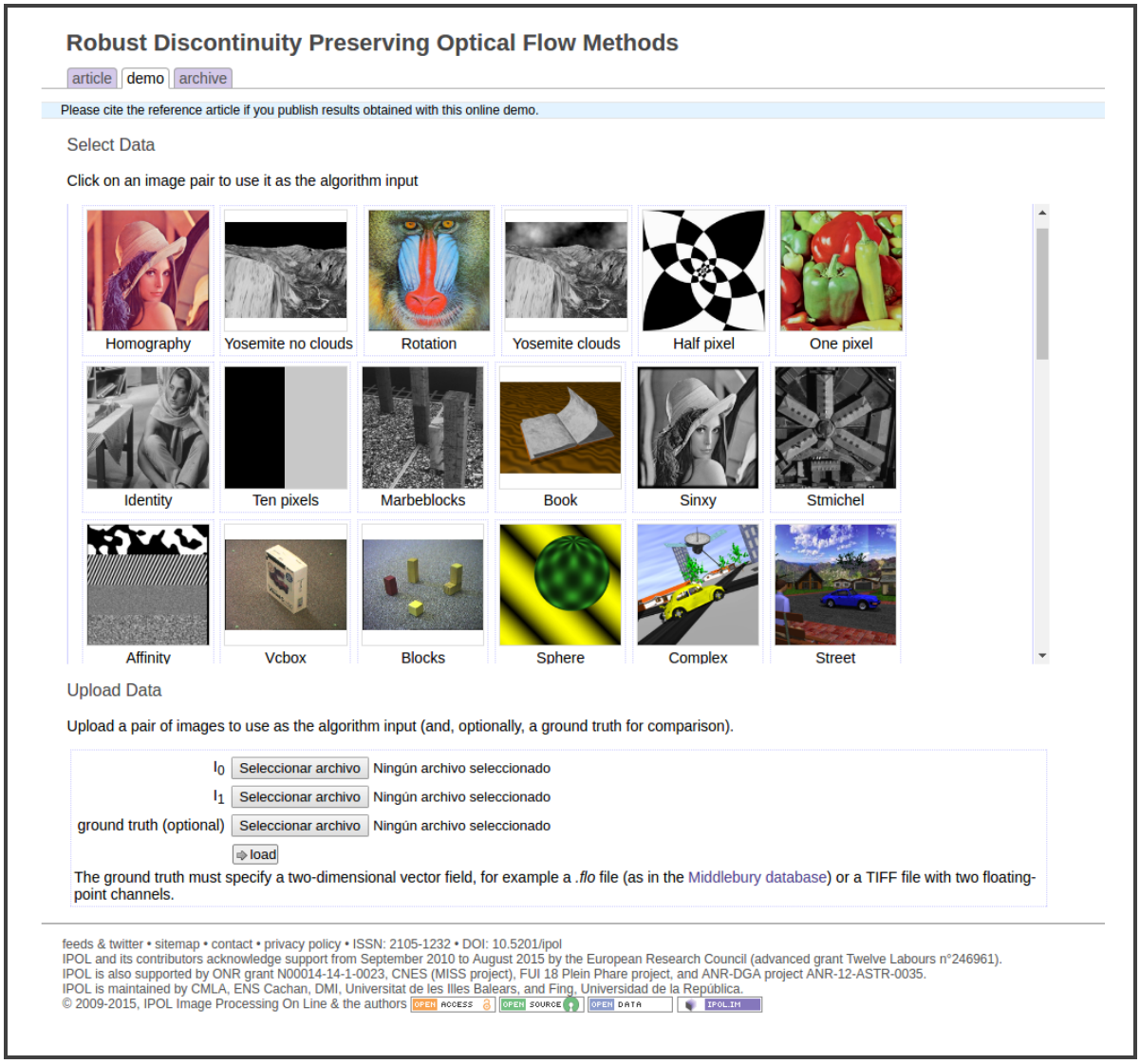


Figure IV-2: IPOL demo for the article. You can choose standard images from several datasets or upload your own pair of images for testing the proposed methods.

Figure IV-3: The parameters page allows us choosing the optical flow approach and the parameters for our experiment.

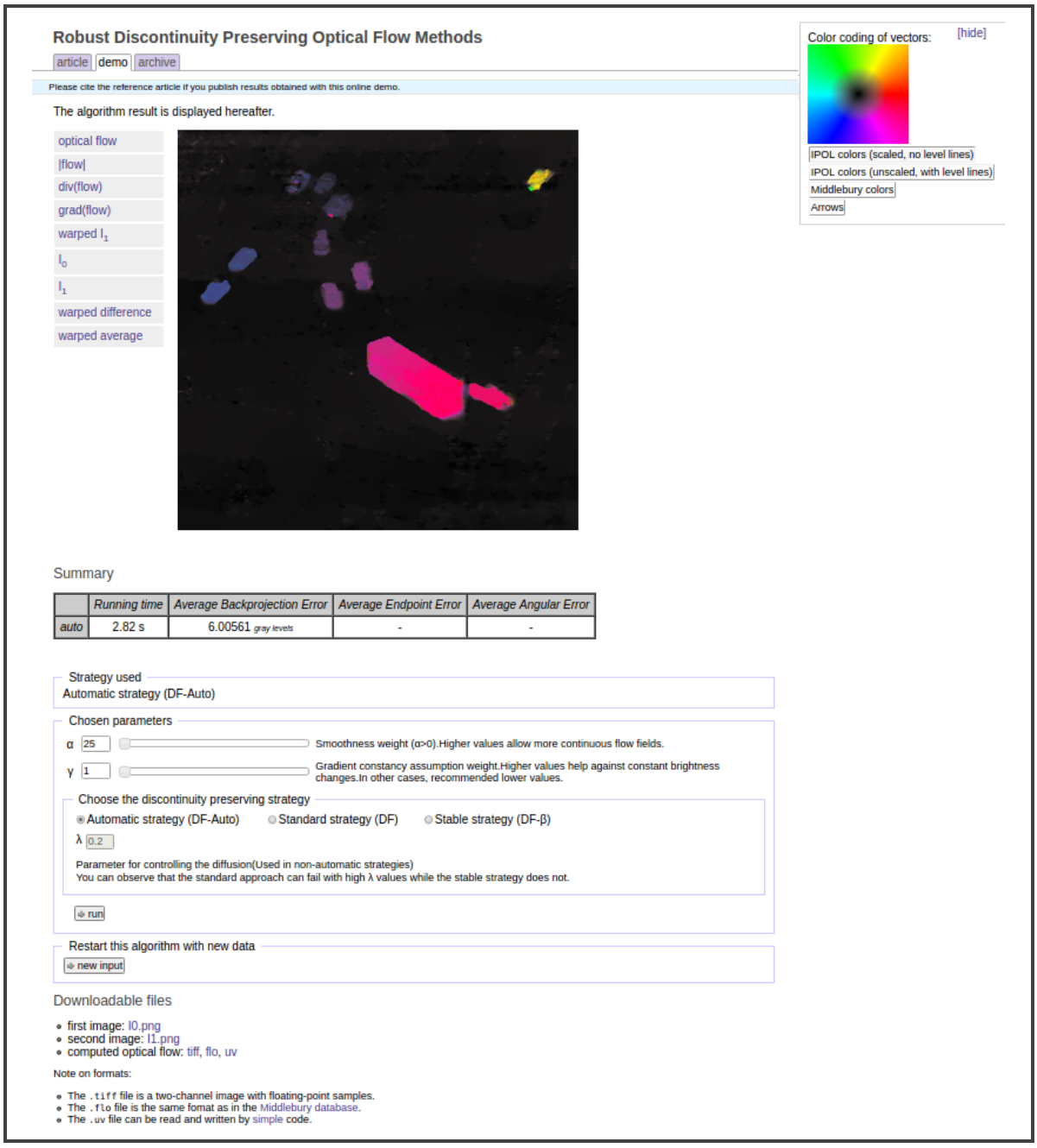


Figure IV-4: Web page with the results achieved by the method. We observe several options for visualizing them. At the upper-right corner, we can select different schemes for the motion representation.

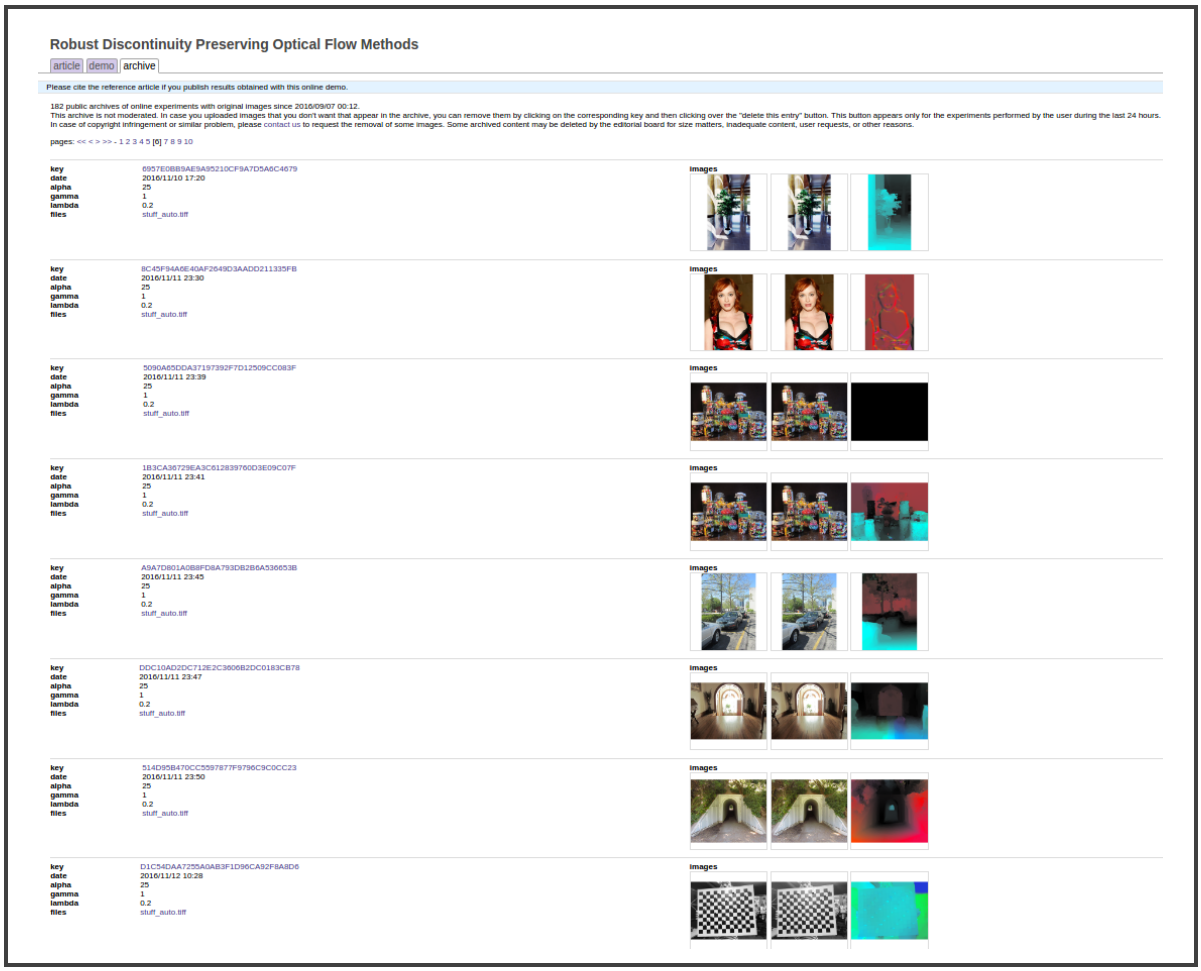


Figure IV-5: Archive page. The IPOL users can upload their own images (with or without true flows), whose results are stored by the demo in the archive.

List of Figures

List of Tables

Bibliography

[Alvarez99] L. Alvarez, J. Esclarín, M. Lefébure, J. Sánchez. A PDE model for computing the optical flow. *XVI Congreso de Ecuaciones Diferenciales y Aplicaciones (CEDYA)*, pags. 1349–1356, Las Palmas de Gran Canaria, Spain, 1999.

[Alvarez00] L. Alvarez, J. Weickert, J. Sánchez. Reliable Estimation of Dense Optical Flow Fields with Large Displacements. *International Journal of Computer Vision*, vol. 39, num. 1, pags. 41–56, 2000.

[Alvarez07] L. Alvarez, R. Deriche, T. Papadopoulo, J. Sánchez. Symmetrical Dense Optical Flow Estimation with Occlusions Detection. *International Journal of Computer Vision*, vol. 75, num. 3, pags. 371–385, 2007.

[Álvarez08] L. Álvarez, C. Castaño, M. García, K. Krissian, L. Mazorra, A. Salgado, J. Sánchez. Multi-Channel Satellite Image Analysis Using a Variational Approach. *Pure and Applied Geophysics*, vol. 165, num. 6, pags. 1071–1093, 2008.

[Anandan89] P. Anandan. A Computational Framework and an Algorithm for the Measurement of Visual Motion. *International Journal of Computer Vision*, vol. 2, pags. 283–310, 1989.

[Andrews03] R. J. Andrews, B. C. Lovell. Color Optical Flow. *In Eds. Proceedings Workshop on Digital Image Computing*, pags. 135–139, 2003.

[Aubert99] G. Aubert, R. Deriche, P. Kornprobst. Computing Optical Flow via Variational Techniques. *SIAM Journal on Applied Mathematics*, vol. 60, pags. 156–182, 1999.

[Ayvaci12] A. Ayvaci, M. Raptis, S. Soatto. Sparse Occlusion Detection with Optical Flow. *International Journal of Computer Vision*, vol. 97, num. 3, pags. 322–338, May 2012.

[Bailer15] C. Bailer, B. Taetz, D. Stricker. Flow Fields: Dense Correspondence Fields for Highly Accurate Large Displacement Optical Flow Estimation. *Computer Vision (ICCV). International Conference on Computer Vision (ICCV-15), December 13-16, Santiago, Chile*. IEEE, 2015.

[Baker07a] S. Baker, D. Scharstein, J. Lewis, S. Roth, M. Black, R. Szeliski. A Database and Evaluation Methodology for Optical Flow. *International Conference on Computer Vision (ICCV 2007)*, 2007.

[Baker07b] S. Baker, D. Scharstein, J. P. Lewis, S. Roth, M. J. Black, R. Szeliski. A Database and Evaluation Methodology for Optical Flow. *International Conference on Computer Vision*, pags. 1–8, 2007.

[Ballester12] C. Ballester, L. Garrido, V. Lazcano, V. Caselles. *A TV-L1 Optical Flow Method with Occlusion Detection*, pags. 31–40. Springer Berlin Heidelberg, Berlin, Heidelberg, 2012.

[Barnes09] C. Barnes, E. Shechtman, A. Finkelstein, D. B. Goldman. PatchMatch: A Randomized Correspondence Algorithm for Structural Image Editing. *ACM Transactions on Graphics (Proc. SIGGRAPH)*, vol. 28, num. 3, , Aug. 2009.

[Barnes10] C. Barnes, E. Shechtman, D. B. Goldman, A. Finkelstein. The Generalized PatchMatch Correspondence Algorithm. *European Conference on Computer Vision*, Sep. 2010.

[Barron94] J. L. Barron, D. J. Fleet, S. S. Beauchemin. Performance of optical flow techniques. *International Journal of Computer Vision*, vol. 12, num. 1, pags. 43–77, 1994.

[Barron01] J. Barron, R. Klette. Experience with Optical Flow in Color Video Image Sequences. *Image and Vision Computing 2001*, pags. 195–200, October 2001.

[Barron02] J. Barron, R. Klette. Quantitative color optical flow. *Pattern Recognition, 2002. Proceedings. 16th International Conference on*, volumen 4, pags. 251–255 vol.4, 2002.

[Battiti91] R. Battiti, E. Amaldi, C.Koch. Computing optical flow across multiple scales: an adaptive coarse-to-fine strategy. *International Journal of Computer Vision*, vol. 6, num. 2, pags. 133–145, 1991.

[Bergen92] J. R. Bergen, P. Anandan, K. J. Hanna, R. Hingorani. Hierarchical model-based motion estimation. *European conference on computer vision*, pags. 237–252. Springer, 1992.

[Bigun87] J. Bigun, G. H. Granlund. Optimal orientation detection of linear symmetry. *Proc. First Int'l Conf. Comput. Vision*, pags. 433–438, 1987.

[Bin10] L. Bin, D. Minghui, H. Jinlong. Color optical flow estimation based on gradient fields with extended constraints. *Networking and Information Technology (ICNIT), 2010 International Conference on*, pags. 279–283, June 2010.

[Black91] M. J. Black, P. Anandan. Robust dynamic motion estimation over time. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pags. 292–302, June 1991.

[Black93] M. J. Black, P. Anandan. A framework for the robust estimation of optical flow. *Proceedings of Fourth International Conference on Computer Vision*, pags. 231–236, 1993.

[Black94] M. J. Black. Recursive non-linear estimation of discontinuous flow fields. *Proceedings of the third European conference on Computer vision (vol. 1)*, ECCV '94, pags. 138–145, Secaucus, NJ, USA, 1994. Springer-Verlag New York, Inc.

[Black96] M. J. Black, P. Anandan. The Robust Estimation of Multiple Motions: Parametric and Piecewise-Smooth Flow Fields. *Computer Vision and Image Understanding*, vol. 63, num. 1, pags. 75 – 104, 1996.

[Black98] M. J. Black, G. Sapiro, D. H. Marimont, D. Heeger. Robust anisotropic diffusion. *Trans. Img. Proc.*, vol. 7, num. 3, pags. 421–432, Mar. 1998.

[Bornemann96] F. Bornemann, P. Deuflhard. The cascadic multigrid method for elliptic problems. *Numerische Mathematik*, vol. 75, pags. 135–152, 1996.

[Brox04] T. Brox, A. Bruhn, N. Papenberg, J. Weickert. High accuracy optical flow estimation based on a theory for warping. *European Conference on Computer Vision (ECCV)*, volumen 3024 de *LNCS*, pags. 25–36, Prague, Czech Republic, May 2004. Springer.

[Brox10] T. Brox, J. Malik. Large Displacement Optical Flow: Descriptor Matching in Variational Motion Estimation. *IEEE PAMI*, vol. 99, num. PrePrints, , 2010.

[Brox11a] T. Brox, J. Malik. Large Displacement Optical Flow: Descriptor Matching in Variational Motion Estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, num. 3, pags. 500–513, March 2011.

[Brox11b] T. Brox, J. Malik. Large displacement optical flow: descriptor matching in variational motion estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, num. 3, pags. 500–513, 2011.

[Bruhn05a] A. Bruhn, J. Weickert. Towards Ultimate Motion Estimation: Combining Highest Accuracy with Real-Time Performance. *International Conference on Computer Vision (ICCV)*, volumen 1, pags. 749–755, Washington, DC, USA, October 2005. IEEE Computer Society. http://dx.doi.org/10.1109/ICCV.2005.240.

[Bruhn05b] A. Bruhn, J. Weickert, C. Schnörr. Lucas /Kanade meets Horn/Schunck: combining local and global optic flow methods. *International Journal of Computer Vision*, vol. 61, pags. 211–231, February 2005.

[Bruhn06] A. Bruhn. *Variational Optic Flow Computation: Accurate Modelling and Efficient Numerics*. Tesis Doctoral, Department of Mathematics and Computer Science, Saarland University, Germany, 2006.

[Butler12] D. J. Butler, J. Wulff, G. B. Stanley, M. J. Black. A naturalistic open source movie for optical flow evaluation. A. Fitzgibbon et al. (Eds.), editor, *European Conf. on Computer Vision (ECCV)*, Part IV, LNCS 7577, pags. 611–625. Springer-Verlag, Oct. 2012.

[Campani90] M. Campani, A. Verri. Computing optical flow from an overconstrained system of linear algebraic equations. *3rd International Conference on Computer Vision*, pags. 22–26, 1990.

[Chambolle04] A. Chambolle. An Algorithm for Total Variation Minimization and Applications. *Journal of Mathematical Imaging and Vision*, vol. 20, num. 1, pags. 89–97, 2004.

[Charbonnier97] P. Charbonnier, L. Blanc-Feraud, G. Aubert, M. Barlaud. Deterministic edge-preserving regularization in computed imaging. *IEEE Transactions on Image Processing*, vol. 6, num. 2, pags. 298–311, Feb 1997.

[Chin94] T. M. Chin, W. C. Karl, A. S. Willsky. Probabilistic and sequential computation of optical flow using temporal coherence. *IEEE Transactions on Image Processing*, vol. 3, num. 6, pags. 773–788, 1994.

[Cohen93] I. Cohen. Nonlinear Variational Method for Optical Flow Computation. *Proceedings of the 8th Scandinavian Conference on Image Analysis*, pags. 523–530, Tromso, Norway, 1993. IAPR.

[Colom15] M. Colom, B. Kerautret, N. Limare, P. Monasse, J.-M. Morel. IPOL: a new journal for fully reproducible research; analysis of four years development. *Reproducibility of Computation Based Research: Languages, Standards, Methodologies and Platforms (NTMS 2015 Reproducibility of Computation Based Research Languages Standards Meth)*, Paris, France, Jul. 2015.

[Dagobert] T. Dagobert. The fabrication of ground truths for evaluating high accurate stereovision algorithms. *Image Processing On Line*, vol. Preprint, .

[Enkelmann88] W. Enkelmann. Investigation of Multigrid Algorithms for the Estimation of Optical Flow Fields in Image Sequences. *Comput. Vis. Graph. Image Process*, vol. 43, pags. 150–177, 1988.

[Fennema79] C. Fennema, W. Thompson. Velocity determination in scenes containing several moving objects. *Computer Graphics and Image Proccesings*, vol. 9, pags. 301–315, 1979.

[Garrido15] L. Garrido, E. M. Kalmoun. A Line Search Multilevel Truncated Newton Algorithm for Computing the Optical Flow. *Image Processing On Line*, vol. 5, pags. 124–138, 2015.

[Golland97] P. Golland, A. M. Bruckstein. Motion from Color. *Computer Vision and Image Understanding*, vol. 68, num. 3, pags. 346–362, 1997.

[Horn81] B. K. P. Horn, B. G. Schunck. Determining Optical Flow. *Artificial Intelligence*, vol. 17, pags. 185–203, 1981.

[Hu16] Y. Hu, R. Song, Y. Li. Efficient Coarse-to-Fine PatchMatch for Large Displacement Optical Flow. *IEEE Conference on Computer Vision and Pattern Recognition*, pags. 5704–5712. IEEE, 2016.

[Lei09] C. Lei, Y.-H. Yang. Optical flow estimation on coarse-to-fine region-trees using discrete optimization. *2009 IEEE 12th International Conference on Computer Vision*, pags. 1562–1569. IEEE, 2009.

[Li09] Y. Li, S. Osher. A new median formula with applications to PDE based denoising. *Commun. Math. Sci.*, vol. 7, num. 3, pags. 741–753, 09 2009.

[Limare11] N. Limare, J.-M. Morel. The IPOL Initiative: Publishing and Testing Algorithms on Line for Reproducible Research in Image Processing. *Procedia Computer Science*, vol. 4, pags. 716 – 725, 2011.

[Limare12] N. Limare. *Reproducible research, software quality, online interfaces and publishing for image processing*. Tesis Doctoral, 2012. Thèse de doctorat dirigée par Morel, Jean-Michel Mathématiques appliquées Cachan, Ecole normale supérieure 2012.

[Lu13] J. Lu, H. Yang, D. Min, M. N. Do. Patch Match Filter: Efficient Edge-Aware Filtering Meets Randomized Search for Fast Correspondence Field Estimation. *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2013.

[Lucas81] B. D. Lucas, T. Kanade. An iterative image registration technique with an application to stereo vision. *Proceedings of the 7th international joint conference on Artificial intelligence - Volume 2*, pags. 674–679, San Francisco, CA, USA, 1981. Morgan Kaufmann Publishers Inc.

[Luettgen94] M. Luettgen, W. Karl, A. Willsky. Efficient multiscale regularization with applications to the computation of optical flow. *IEEE Transactions on Image Processing*, vol. 3, num. 1, pags. 41–64, 1994.

[Maintz98] J. Maintz, M. Viergever. A survey of medical image registration. *Medical Image Analysis*, vol. 2, num. 1, pags. 1–36, 1998.

[Markandey90] V. Markandey, B. Flinchbaugh. Multispectral constraints for optical flow computation. *Computer Vision, 1990. Proceedings, Third International Conference on*, pags. 38–41, Dec 1990.

[Mémin98] E. Mémin, P. Pérez. Dense Estimation and Object-Based Segmentation of the Optical-Flow with Robust Techniques. *IEEE Transactions on Image Processing*, vol. 7, num. 5, pags. 703–719, May 1998.

[Mémin02] E. Mémin, P. Pérez. Hierarchical Estimation and Segmentation of Dense Motion Fields. *International Journal of Computer Vision*, vol. 46, num. 2, pags. 129–155, 2002.

[Mileva07] Y. Mileva, A. Bruhn, J. Weickert. Illumination-Robust Variational Optical Flow with Photometric Invariants. F. Hamprecht, C. Schnörr, B. Jähne, editors, *Pattern Recognition*, volumen 4713 de *Lecture Notes in Computer Science*, pags. 152–162. Springer Berlin Heidelberg, 2007.

[Mitiche87] A. Mitiche, Y. F. Wang, J. K. Aggarwal. Experiments in computing optical flow with the gradient-based, multiconstraint method. *Pattern Recognition*, vol. 20, num. 2, pags. 173–179, 1987.

[ML13] E. Meinhardt-Llopis, J. Sánchez, D. Kondermann. Horn-Schunck optical flow with a multi-scale strategy. *Image Processing On Line*, vol. 2013, pags. 151–172, 2013. http://dx.doi.org/10.5201/ipol.2013.20.

[Monzón13] N. Monzón, J. Sánchez, A. Salgado. Optic Flow: Improving Discontinuity Preserving. *14th International Conference on Computer Aided Systems*

Theory (EUROCAST), pags. 114–116, Berlin, Heidelberg, 2013. Springer-Verlag.

[Monzón14a] N. Monzón, J. Sánchez, A. Salgado. Efficient Mechanism for Discontinuity Preserving in Optical Flow Methods. A. Elmoataz, O. Lezoray, F. Nouboud, D. Mammass, editors, *Image and Signal Processing*, volumen 8509 de *Lecture Notes in Computer Science*, pags. 425–432. Springer International Publishing, 2014.

[Monzón14b] N. Monzón, J. Sánchez, A. Salgado. Implementation of a Robust Optical Flow Method for Color Images. Technical Report 4, Universidad de Las Palmas de Gran Canaria, September 2014.

[Monzón16a] N. Monzón, A. Salgado, J. Sánchez. Regularization Strategies for Discontinuity-Preserving Optical Flow Methods. *IEEE Transactions on Image Processing*, vol. 25, num. 4, pags. 1580–1591, April 2016.

[Monzón16b] N. Monzón, A. Salgado, J. Sánchez. Robust Discontinuity Preserving Optical Flow Methods. *Image Processing On Line*, vol. 6, pags. 165–182, 2016.

[Nagel86] H.-H. Nagel, W. Enkelmann. An investigation of smoothness constraints for the estimation of displacement vector fields from image sequences. *IEEE Transanctions on Pattern Analysis and Machine Intelligence*, vol. 8, pags. 565–593, September 1986.

[Nagel90a] H. H. Nagel. Extending the 'oriented smoothness constraint' into the temporal domain and the estimation of derivatives of optical flow. *European Conference on Computer Vision (ECCV) '90, Lecture Notes in Computer Science*, volumen 427, pags. 139–148, 1990.

[Nagel90b] H.-H. Nagel. Extending the 'oriented smoothness constraint' into the temporal domain and the estimation of derivatives of optical flow. *Proceedings of the first European Conference on Computer vision*, ECCV 90, pags. 139–148, New York, NY, USA, 1990. Springer-Verlag New York, Inc.

[Ohta89] N. Ohta. Optical Flow Detection by Color Images. *IEEE International Conference on Image Processing*, , pags. 801–805, 1989.

[Palomares16] R. P. Palomares, E. Meinhardt-Llopis, C. Ballester, G. Haro. FALDOI: A new minimization strategy for large displacement variational optical flow. *ArXiv e-prints*, , , Feb. 2016.

[Papenberg06] N. Papenberg, A. Bruhn, T. Brox, S. Didas, J. Weickert. Highly Accurate Optic Flow Computation with Theoretically Justified Warping. *International Journal of Computer Vision*, vol. 67, num. 2, pags. 141–158, April 2006.

[Perona90] P. Perona, J. Malick. Scale-Space and Edge Detection Using Anisotropic Diffusion. *IEEE Transactions on Pattern Analasys and Machine Intelligence*, vol. 12, pags. 629–629, July 1990.

[Proesmans94] M. Proesmans, L. V. Gool, E. Pauwels, A. Oosterlinck. Determination of optical flow and its discontinuities using non-linear diffusion. *third European Conference on Computer Vision*, volumen 2, pags. 295–304, Secaucus, NJ, USA, 1994. Springer-Verlag.

[Revaud15] J. Revaud, P. Weinzaepfel, Z. Harchaoui, C. Schmid. EpicFlow: Edge-Preserving Interpolation of Correspondences for Optical Flow. *arXiv preprint arXiv:1501.02565*, , , 2015.

[Rudin92] L. I. Rudin, S. Osher, E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D*, vol. 60, pags. 259–268, November 1992.

[Salgado06] A. Salgado, J. Sánchez. A Temporal Regularizer for Large Optical Flow Estimation. *IEEE International Conference on Image Processing (ICIP)*, pags. 1233–1236, 2006.

[Sánchez12] J. Sánchez, N. Monzón, A. Salgado. Parallel Implementation of a Robust Optical Flow Technique. Technical Report 1, Universidad de Las Palmas de Gran Canaria, March 2012.

[Sánchez13a] J. Sánchez, E. Meinhardt-Llopis, G. Facciolo. TV-L1 Optical Flow Estimation. *Image Processing On Line*, vol. 2013, pags. 137–150, 2013.

[Sánchez13b] J. Sánchez, N. Monzón, A. Salgado. Robust Optical Flow Estimation. *Image Processing On Line*, vol. 2013, pags. 252–270, 2013. http://dx.doi.org/10.5201/ipol.2013.21.

[Sánchez13c] J. Sánchez, A. Salgado, N. Monzón. Computing Inverse Optical Flow. Technical Report 3, Universidad de Las Palmas de Gran Canaria, November 2013.

[Sánchez13d] J. Sánchez, A. Salgado, N. Monzón. Direct Estimation of the Backward Flow. *International Conference on Computer Vision Theory and Applications (VISAPP)*, pags. 268–274. Institute for Systems and Technologies of Information, Control and Communication, 2013.

[Sánchez13e] J. Sánchez, A. Salgado, N. Monzón. An Efficient Algorithm for Estimating the Inverse Optical Flow. *6th Iberian Conference on Pattern Recognition and Image Analysis (IbPRIA)*, pags. 390–397. Springer-Verlag, 2013.

[Sánchez13f] J. Sánchez, A. Salgado, N. Monzón. Optical Flow Estimation with Consistent Spatio-temporal Coherence Models. *International Conference on Computer Vision Theory and Applications (VISAPP)*, pags. 366–370. Institute for Systems and Technologies of Information, Control and Communication, 2013.

[Sánchez14] J. Sánchez, A. Salgado, N. Monzón. Preserving Accurate Motion Countours with Reliable Parameter Selection. *IEEE International Conference on Image Processing (ICIP2014)*, pags. 209–213. IEEE International Conference on Image Processing (ICIP2014), 2014.

[Schnörr94] C. Schnörr. Segmentation of visual motion by minimising convex non-quadratic functionals. *IEEE Computer Society Press*, vol. A, pags. 661–663, 1994.

[Shafer85] S. A. Shafer. Using color to separate reflection components. *Color Research & Application*, vol. 10, num. 4, pags. 210–218, 1985.

[Singh91] A. Singh. Incremental Estimation of Image-Flow Using a Kalman Filter. *Visual Motion*, pags. 36–43, 1991.

[Sun03] J. Sun, N.-N. Zheng, H.-Y. Shum. Stereo Matching Using Belief Propagation. *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, num. 7, pags. 787–800, Jul. 2003.

[Sun08] D. Sun, J. P. Lewis, M. J. Black. Learning optical flow. *European Conference on Computer Vision (ECCV)*, pags. 83–97, 2008.

[Sun10a] D. Sun, S. Roth, M. J. Black. Secrets of optical flow estimation and their principles. *IEEE Conference on Computer Vision and Pattern Recognition*, pags. 2432–2439, Los Alamitos, CA, USA, 2010. IEEE Computer Society.

[Sun10b] D. Sun, E. Sudderth, M. J. Black. Layered Image Motion with Explicit Occlusions, Temporal Consistency, and Depth Ordering. *Advances in Neural Information Processing Systems 23*, volumen 23, pags. 2226–2234, 2010.

[Sun12] D. Sun, E. Sudderth, M. J. Black. Layered segmentation and optical flow estimation over time. *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pags. 1768–1775. IEEE, 2012.

[Sánchez15] J. Sánchez, A. Salgado, N. Monzón. Computing inverse optical flow. *Pattern Recognition Letters*, vol. 52, pags. 32 – 39, 2015.

[Tomasi98] C. Tomasi, R. Manduchi. Bilateral Filtering for Gray and Color Images. *ICCV*, pags. 839–846, 1998.

[Tretiak84] O. Tretiak, L. Pastor. Velocity estimation from image sequences with second order differential operators. *Proc. 7th International Conference on Pattern Recognition*, vol. , pags. 20–22, 1984.

[Unger12] M. Unger, M. Werlberger, T. Pock, H. Bischof. Joint motion estimation and segmentation of complex scenes with label costs and occlusion modeling. *CVPR*, pags. 1878–1885. IEEE, 2012.

[Uras88] S. Uras, F. Girosi, A. Verri, V. Torre. A computational approach to motion perception. *Biological Cybernetics*, vol. 60, pags. 79–97, 1988.

[vdW04] J. van de Weijer, T. Gevers. Robust optical flow from photometric invariants. *Image Processing, 2004. ICIP '04. 2004 International Conference on*, volumen 3, pags. 1835–1838 Vol. 3, Oct 2004.

[Volz11] S. Volz, A. Bruhn, L. Valgaerts, H. Zimmer. Modeling Temporal Coherence for Optical Flow. *International Conference on Computer Vision*, November 2011.

[Wedel09a] A. Wedel, D. Cremers, T. Pock, H. Bischof. Structure- and motion-adaptive regularization for high accuracy optic flow. *Proceedings of IEEE International Conference on Computer Vision*, pags. 1663–1668, Sep. 2009.

[Wedel09b] A. Wedel, T. Pock, C. Zach, H. Bischof, D. Cremers. An improved algorithm for TV-L1 optical flow. *Statistical and Geometrical Approaches to Visual Motion Analysis*, pags. 23–45. Springer Berlin Heidelberg, 2009.

[Weickert97] J. Weickert. A Review of Nonlinear Diffusion Filtering. *Proceedings of the First International Conference on Scale-Space Theory in Computer Vision*, SCALE-SPACE '97, pags. 3–28, London, UK, UK, 1997. Springer-Verlag.

[Weickert98] J. Weickert. *Anisotropic diffusion in image processing*, volumen 1. Teubner Stuttgart, 1998.

[Weickert01] J. Weickert, C. Schnörr. Variational Optic Flow Computation with a Spatio-Temporal Smoothness Constraint. *Journal of Mathematical Imaging and Vision*, vol. 14, num. 3, pags. 245–255, May 2001.

[Weinzaepfel13] P. Weinzaepfel, J. Revaud, Z. Harchaoui, C. Schmid. Deepflow: Large displacement optical flow with deep matching. *Computer Vision (ICCV), 2013 IEEE International Conference on*, pags. 1385–1392. IEEE, 2013.

[Werlberger09] M. Werlberger, W. Trobin, T. Pock, A. Wedel, D. Cremers, H. Bischof. Anisotropic Huber-L1 Optical Flow. *BMVC*. British Machine Vision Association, 2009.

[Werlberger10] M. Werlberger, T. Pock, H. Bischof. Motion estimation with non-local total variation regularization. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pags. 2464–2471, San Francisco, CA, USA, June 2010.

[Wohn83] K. Wohn, L. Davis, P. Thrift. Motion estimation based on multiple local constraints and nonlinear smoothing. *Pattern Recognition*, vol. 16, num. 6, pags. 563–570, 1983.

[Xiao06] J. Xiao, H. Cheng, H. Sawhney, C. Rao, M. Isnardi. Bilateral filtering-based optical flow estimation with occlusion detection. *European Conference on Computer Vision*, volumen I, pags. 211–224, 2006.

[Xu10] L. Xu, J. Jia, Y. Matsushita. Motion Detail Preserving Optical Flow Estimation. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pags. 1293–1300, June 2010.

[Xu12] L. Xu, J. Jia, Y. Matsushita. Motion Detail Preserving Optical Flow Estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, num. 9, pags. 1744–1757, Sep. 2012.

[Yedidia03] J. S. Yedidia, W. T. Freeman, Y. Weiss. Understanding belief propagation and its generalizations. *Exploring artificial intelligence in the new millennium*, vol. 8, pags. 236–239, 2003.

[Yoon06] K.-J. Yoon, I.-S. Kweon. Adaptive Support-Weight Approach for Correspondence Search. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, num. 4, pags. 650–656, 2006.

[Zach07] C. Zach, T. Pock, H. Bischof. A Duality Based Approach for Realtime TV-L1 Optical Flow. *Pattern Recognition*, volumen 4713 de *Lecture Notes in Computer Science*, chapter 22, pags. 214–223. Springer Berlin Heidelberg, Berlin, Heidelberg, 2007.

[Zimmer09] H. Zimmer, A. Bruhn, J. Weickert, L. Valgaerts, A. Salgado, B. Rosenhahn, H.-P. Seidel. Complementary Optic Flow. *Proceedings of the 7th*

International Conference on Energy Minimization Methods in Computer Vision and Pattern Recognition, EMMCVPR '09, pags. 207–220, Berlin, Heidelberg, 2009. Springer-Verlag.

[Zimmer11] H. Zimmer, A. Bruhn, J. Weickert. Optic Flow in Harmony. *International Journal of Computer Vision*, vol. 93, num. 3, pags. 368–388, Jul. 2011.

[Zitová03] B. Zitová, J. Flusser. Image registration methods: A survey. *Image and Vision Computing*, vol. 21, num. 11, pags. 977–1000, 2003.