Regularizing a Set of Unstructured 3D Points from a Sequence of Stereo Images

Luis Alvarez, Carmelo Cuenca, and Javier Sánchez

Departamento de Informática y Sistemas
Universidad de Las Palmas de G.C.
Campus Universitario de Tafira
35017, Las Palmas
{lalvarez,ccuenca,jsanchez}@dis.ulpgc.es
http://serdis.dis.ulpgc.es/~{lalvarez,jsanchez}

Abstract. In this paper we present a method for the regularization of a set of unstructured 3D points obtained from a sequence of stereo images. This method takes into account the information supplied by the disparity maps computed between pairs of images to constraint the regularization of the set of 3D points. We propose a model based on an energy which is composed of two terms: an attachment term that minimizes the distance from 3D points to the projective lines of camera points, and a second term that allows for the regularization of the set of 3D points by preserving discontinuities presented on the disparity maps. We embed this energy in a 2D finite element method. After minimizing, this method results in a large system of equations that can be optimized for fast computations. We derive an efficient implicit numerical scheme which reduces the number of calculations and memory allocations.

1 Introduction

This paper deals with the problem of 3D geometry reconstruction from multiple 2D views. Recently, a new accurate technique based on a variational approach has been proposed in [7],[8]. Using a level set approach, this technique optimizes a 3D surface by minimizing an energy that takes into account the surface regularity as well as the projection of the surface on different images.

In this paper we propose a different approach which is also based on a variational formulation but only using a disparity estimation between images and without defining explicitly any 3D surface. We will assume that the cameras are calibrated in the strong sense (see [6], [9] or [11] for more details). In the last years, very accurate techniques to estimate the disparity map in a stereo pair of images have been proposed. To extend these techniques to the case of multiple views is not a trivial problem. Roughly speaking the 3D geometry estimation that we propose can be divided in the following steps:

- For each pair of consecutive images, we estimate a dense disparity map using the accurate technique developed in [1]. We estimate such disparity

  map forward and backward, that is, from one image to the next one and in
  the opposite direction.
- We estimate sequences of corresponding points across the multiple view im-
  age sequence. Basically, we try to connect points between images following
  the disparity map estimations. We select sequences of correspondent points
  for which the forward and backward disparity estimations are coherent.
- From each selected corresponding points sequence we recover a 3D point by
  intersecting the projection lines of the points in the sequence. By collecting
  the 3D points obtained from each sequence we recover an unstructured set
  of 3D points.
- Typically, the recovered set of 3D points is noisy, because of errors in the
  camera calibration process, errors in the disparity estimations, errors in the
  corresponding point sequences computations, etc., so some kind of regulariza-
  tion is needed. In this paper, we propose a new variational model to smooth
  the unstructured set of 3D points. This regularization model is based on the
  2D image information and does not require to define any kind of geometric
  relation between the 3D points.

The proposed technique provides a smooth set of unstructured 3D points.
In this paper we do not address the problem of defining one or several surfaces
fitting the set of points. Such surfaces could be recovered using some standard
methods like Alpha shapes [5], Ball pivot [4] or Voronoi filtering [3]. We notice
that most of such techniques require the collection of points to be smooth enough
to recover the surface. So the regularization step we propose is necessary to
improve the results of such techniques.

The regularization model we propose, which is the main contribution of the
paper is based on a variational approach. This model is designed in order to
maintain the final 3D regularized surface next to the original surface and also
to enable a regularization by preserving discontinuities on the disparity maps.
The regularization is carried out by means of an operator which is similar to the
Nagel–Enkelmann's operator [12]. This operator has already proven its efficiency
in other fields like stereoscopic reconstruction [1], optical flow estimation [2],
etc. We have modified this operator to include the information given by the
forward and backward optical flows of every camera. In our case it is convenient
to preserve these discontinuities because disparity maps represent the depth
variation of the set of 3D points.

For every camera we will have a set of 2D projection points (extracted from
the selected sequences of points described above) forming a grid. These grids
are not necessary square and equally distributed as we would expect for regular
images. On the contrary we will have in most cases some distributed points on
the cameras with float precision. For this, we embed the energy in a finite element
method and express the set of 3D points in terms of some basis functions. We
will use a Delaunay triangulation for every camera and every point on a mesh
will have a correspondent 3D point.

Deriving this energy and searching for the minimum yields a system of as
many equations as 3D points are there in the set. The system matrix is a sparse

$h_-^a(x,y)$ $\mathcal{C}_1$ $h_+^a(x,y)$

$\mathcal{C}_0$ $h_+^{c_0}(x,y)$ (x_{c_1}, y_{c_1}) $h_-^{c_2}(x,y)$ $\mathcal{C}_2$

(x_{c_0}, y_{c_0}) $\mathbf{F^1}$ (x_{c_2}, y_{c_2})

$\mathbf{F^0}$ $\mathbf{F^2}$

$R^{c_1}(\overline{X})$

$R^{c_0}(\overline{X})$ $R^{c_2}(\overline{X})$

$\mathbf{X} = (X^0, X^1, X^2)$
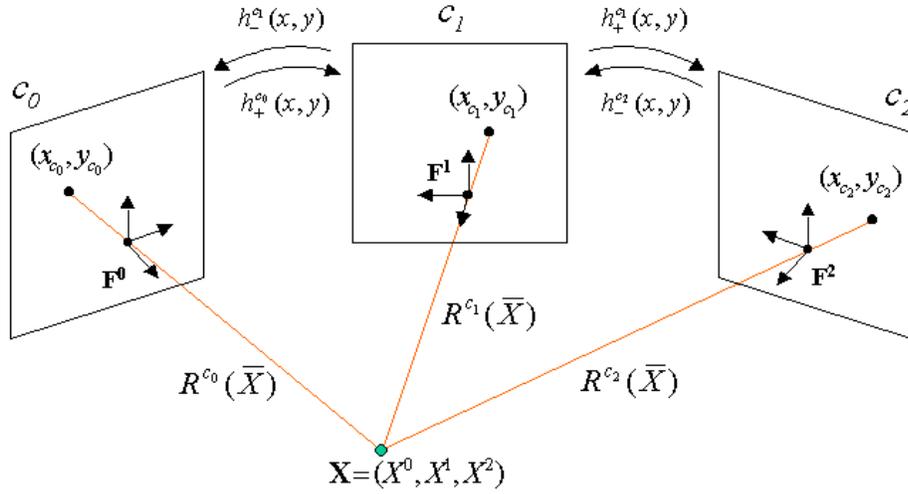
Fig. 1. Notation

matrix that will only have no-null values in some special locations. To solve the system we first arrange the matrices to optimize their size in memory and then implement a Gauss-Seidel numerical scheme that is very efficient in the number of calculations.

Finally we present two experimental results to evaluate the method. In the first experiment we use a single stereo pair of a face. We show the stereo pair and the reconstruction obtained from the method explained in [1]. Then we show the Delaunay triangulations of both cameras and finally some results of applying our regularizing method with different values for the parameters. In the other experiment we use larger sequences of stereo images. We show different results from several points of view.

In Sect. 2 the notation is explained. In Sect. 3 we study the model proposing an energy with the attachment and regularizing term and explaining the operator derived from the Nagel–Enkelmann operator. In Sect. 4 we explain how to embed the method into the 2D finite element approach. In Sect. 5 we present the numerical experiments.

2   Notation

As we will see later this method involves complex notation that is mainly affected by three different aspects: we use some notation for the camera configuration, notation for the stereo matching process and notation for the representation of the 3D points. We will focus on Fig. (1) to explain all these aspects.

We start explaining the camera notation. We suppose there is a set of projective cameras, c_i $(i = 1, .., N_c)$, N_c denoting the number of cameras. The projec-

tion model assumed for the cameras is the projective one. This model projects a 3D point $\mathbf{X} = [(X^0, X^1, X^2)]^t$ to a 2D image point $\mathbf{x_c} = [x_c, y_c]^t$ through a 3×4 projection matrix $\mathbf{P_c}$ via $s\hat{\mathbf{x}}_\mathbf{c} = \mathbf{P_c}\hat{\mathbf{X}}$, where s is a nonzero scale factor and the notation $\hat{\mathbf{p}}$ is such that if $\mathbf{p} = [x_1, x_2, \ldots, x_n]^t$ then $\hat{\mathbf{p}} = [x_1, x_2, \ldots, x_n, 1]^t$.

We note by $\{\mathbf{X_j}\}_{j=0,\ldots,N_p-1}$ the unstructured set of 3D points, where N_p is the number of points considered. Each point $\mathbf{X_j}$ is projected in the camera system following a sequence of corresponding points. We note by

$$(x_{c_i}(\mathbf{X_j}), y_{c_i}(\mathbf{X_j})) \quad i = 0, ..., N(\mathbf{X_j}) - 1 \tag{1}$$

the sequence of corresponding points associated to the 3D point X_j. $N(\mathbf{X_j})$ represents the number of cameras where X_j is projected. We also note by $N_p(c)$ the number of selected sequences of corresponding points passing through camera c.

We may compute from the projection matrices $\mathbf{P_c}$(see [6]) the focuses $\mathbf{F}^c = (F^{c,0}, F^{c,1}, F^{c,2})$ for every camera c. For every 2D projection $\mathbf{x_c}$ and for every camera c we have an unitary vector $\bar{l}^c(x,y) = (l^{c,0}(x,y), l^{c,1}(x,y), l^{c,2}(x,y))$ in the direction of focus $\mathbf{F}^c$. We denote by $\bar{R}^c(x,y)$ the line passing through focus $\mathbf{F}^c$ and having $\bar{l}^c(x,y)$ as directional vector.

Let us now talk about the notation for the stereo aspects. For every pair of consecutive cameras on the sequence we have the estimated optical flow from camera c into the previous camera $c-1$, $\bar{h}^c_-(x,y) = (u^c_-(x,y), v^c_-(x,y))$, and the corresponding flow from camera c into the following, $\bar{h}^c_+(x,y) = (u^c_+(x,y), v^c_+(x,y))$.

For every camera c we define a function $\bar{X}^c : \mathbb{R}^2 \longrightarrow \mathbb{R}^3$ that puts in correspondence bi-dimensional points (x,y) on the projection plane of camera c into 3D points $(X^{c,0}(x,y), X^{c,1}(x,y), X^{c,2}(x,y))$. Let us call $\mathbf{X} = (X^0, X^1, X^2)$ the 3D point obtained by the minimum distance to the set of lines $\bar{R}^c(x,y)$ that passes through focus $\mathbf{F}^c$ and has $\bar{l}^c(x,y)$ as directional vector for the cameras in where there is a correspondent projection. Suppose that every 3D point is visible from $N(\mathbf{X})$ cameras. These cameras are denoted by $c_i(\mathbf{X})$ for $i = 0, .., N(\mathbf{X}) - 1$. The pixel coordinates for $\mathbf{X}$ on camera c_i is denoted by $(x_{c_i}(\mathbf{X}), y_{c_i}(\mathbf{X}))$. Therefore we have

$$\mathbf{X} = \bar{X}^c (x_{c_i}(\mathbf{X}), y_{c_i}(\mathbf{X})) \quad i = 0, ..., N(\mathbf{X}) - 1 . \tag{2}$$

Finally, the derivatives of a function f are represented as $\frac{df}{dx} = f_x$ and $\frac{df}{dy} = f_y$ and ∇X is the gradient of function X.

3   The 3D Regularizing Model

We tackle the problem of 3D regularizing by means of a variational approach in where the solution for the minimization of a global energy is the regularized set of 3D points.

Our method regularizes a set of 3D points by constraining the process with the information given by the optical flows computed for every camera.

The energy to be minimized for the regularization of the set of 3D point is:

$$E\left(\bar{X}^0, .., \bar{X}^{N_c-1}\right) = \sum_{c=0}^{N_c-1}\left(\int_\Omega dist(\bar{X}^c, \bar{R}^c)^2 + \alpha\sum_{i=0}^{2}\int_\Omega \left(\nabla X^{c,i}\right)^T D(\bar{h}^c)\nabla X^{c,i}\right)$$
(3)

where α is a parameter that states the balance between the two terms and $dist(\bar{X}^c, \bar{R}^c)$ denotes the distance from point $\bar{X}^c$ to the straight line $\bar{R}^c$ and is given by formula

$$dist(\bar{X}^c, \bar{R}^c)^2 = \sum_{i=0}^{2}\left(X^{c,i} - F^{c,i}\right)^2 - \left(\sum_{i=0}^{2}l^{c,i}\left(X^{c,i} - F^{c,i}\right)\right)^2.$$
(4)

Our unknowns are $\left(\bar{X}^0, ..., \bar{X}^{N_c-1}\right)$ for cameras $0, ..., N_c - 1$ respectively. We suppose every camera has its own set of 3D points but these sets are referred to a common global set of 3D points. For consecutive cameras a large part of their set of points are going to be coincident. In particular the functions X^c are related by (2). We will see later on Sect. 4 that formulating the energy in this way allows us to create a numerical scheme by summing the contribution of every camera.

The first term of (3) minimizes the distance from the 3D point $\bar{X}^c$ to the straight lines generated by the corresponding camera points passing through the camera focus. The second is a smoothness term that minimizes the variation of the 3D points according to the information given by the gradient of $\bar{h}^c_+$ and $\bar{h}^c_-$. In this case we minimize the variation of the 3D surface by using an operator which is very similar to the Nagel–Enkelmann operator. This operator allows the method to regularize isotropically when the 3D points set varies smoothly and anisotropically when there is a strong variation of the disparity map, thus, respecting regions of different depths.

Matrix $D(\bar{h}^c)$ is given by the following expression:

$$D(\bar{h}^c) = \frac{M(\bar{h}^c_+) + M(\bar{h}^c_-)}{\left\|\nabla\bar{h}^c_+\right\|^2 + \left\|\nabla\bar{h}^c_-\right\|^2}$$
(5)

where

$$\left\|\nabla\bar{h}\right\|^2 = \left\|\nabla u\right\|^2 + \left\|\nabla v\right\|^2$$

and

$$M(\bar{h}) = \begin{cases} \begin{pmatrix} \frac{\partial u}{\partial y}^2 + \frac{\partial v}{\partial y}^2 & -\left(\frac{\partial u}{\partial y}\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}\frac{\partial v}{\partial x}\right) \\ -\left(\frac{\partial u}{\partial y}\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}\frac{\partial v}{\partial x}\right) & \frac{\partial u}{\partial x}^2 + \frac{\partial v}{\partial x}^2 \end{pmatrix} & if\ \lambda^2 \geq \left\|\nabla\bar{h}\right\|^2 \\ \begin{pmatrix} \frac{1}{2} & 0 \\ 0 & \frac{1}{2} \end{pmatrix} & if\ \lambda^2 < \left\|\nabla\bar{h}\right\|^2 \end{cases}.$$
(6)

This is a projection matrix on the orthogonal space to the vector field $h^c_{+/-}$. This matrix is similar to the Nagel-Enkelmann operator [12] but instead of using

λ within the matrix we use it as a real threshold in a non-continuous function. We also consider the optical flow as the projection space and take advantage of the information in both directions. λ is a threshold that states the contour value from where we obtain an anisotropic behavior. This parameter is obtained through an isotropy fraction, s, as is explained in [1]. When $s \to 0$ the method becomes anisotropical and when $s \to 1$ the method becomes isotropical.

The first problem that comes out when we discretize this energy is that in every camera, the point distribution $(x_c(\bar{X}), y_c(\bar{X}))$ is given in float precision and, therefore, it does not consist of a simple pixel grid where we may discretize the equations using the surrounding neighbors. To solve this problem we use a finite element method.

4   Finite Elements Method

We are going to discretize the function X^c using the discrete set of points $\{\mathbf{X_j}\}_{j=0,..,N_p-1}$ obtained from the sequence of 2D corresponding image points $(x_{c_i}(\mathbf{X_j}), y_{c_i}(\mathbf{X_j}))$, $i = 0, ..., N(\mathbf{X_j})-1$. Since $(x_{c_i}(\mathbf{X_j}), y_{c_i}(\mathbf{X_j}))$ are given in float precision and they are distributed in the images in an heterogeneous way, it is not convenient to use a classical finite difference scheme. So we propose a finite element scheme that takes into account the geometry of the set $(x_{c_i}(\mathbf{X_j}), y_{c_i}(\mathbf{X_j}))$ in every image.

Following the notation introduced in Sect. 2 let us call $N_p(c)$ the number of 3D points visible from camera c. For any of these points $(l = 0, .., N_p(c)-1)$, $j(l)$ denote the associated 3D point index in a global list of 3D points. We generate, using a Delaunay triangulation (see [10]), a grid of K_n^c triangles in every camera c with $(x_c(\bar{X}_{j(l)}), y_c(\bar{X}_{j(l)}))_{l=0,...,N_p(c)-1}$ points. We will call $N_t(c)$ the number of triangles generated on camera c.

In the finite element approach the $X^{c,i}(x,y)$ functions are approximated as

$$X^{c,i}(x,y) = \sum_{l=0}^{N_p(c)-1} X_{j(l)}^{c,i} \phi_l^c(x,y) \tag{7}$$

where $X_{j(l)}^{c,i}$ are the unknowns of our system and $\phi_{j(l)}^c(x,y)$ is a set of basis functions centered on the nodes $(x_c(\bar{X}_{j(l)}), y_c(\bar{X}_{j(l)}))$. These basis functions are defined over each triangle K_n^c of vertexes $(j(l), j(m), j(n))$ in the following way:

$$Dx(\tau, \upsilon) = (x_c(\bar{X}_{j(\tau)}) - x_c(\bar{X}_{j(\upsilon)}))$$
$$Dy(\tau, \upsilon) = (y_c(\bar{X}_{j(\tau)}) - y_c(\bar{X}_{j(\upsilon)}))$$

$$\phi_l^c(x,y) = \frac{(y - y_c(\bar{X}_{j(m)}))Dx(n,m) - (x - x_c(\bar{X}_{j(m)}))Dy(n,m)}{Dy(l,m)Dx(n,m) - Dx(l,m)Dy(n,m)} . \tag{8}$$

Note that the gradient of ϕ on the triangle is constant and is given by

$$\nabla \phi_l^c(x,y) = \begin{pmatrix} \frac{-Dy(n,m)}{Dy(l,m)Dx(n,m)-Dx(l,m)Dy(n,m)} \\ \frac{Dx(n,m)}{Dy(l,m)Dx(n,m)-Dx(l,m)Dy(n,m)} \end{pmatrix} . \tag{9}$$

First, we note that $\bar{X}_{j(l)}^c$ corresponds to a 3D point visible from camera c and that such a point references a 3D point, $\bar{X}_j$, of the global 3D point list, that is, the true unknowns are the $\bar{X}_j$ points. Replacing $X^{c,i}(x,y) = \sum_{l=0}^{N_c-1} X_{j(l)}^{c,i} \phi_{j(l)}(x,y)$ in (3) we obtain:

$$E\left(\{\bar{X}_j\}_{j=0,..,N_p-1}\right) = \sum_{c=0}^{N_c-1} \sum_{l=0}^{N_p(c)-1} dist(\bar{X}_{j(l)}^c, \bar{R}^c)^2$$

$$+ \alpha \sum_{c=0}^{N_c-1} \sum_{i=0}^{2} \sum_{n=0}^{N_t(c)-1} \sum_{l,l'=0}^{N_p(c)-1} X_{j(l)}^{c,i} X_{j(l')}^{c,i} \int_{K_n^c} \left(\nabla\phi_{j(l)}^c\right)^T D(u^c, v^c)\nabla\phi_{j(l')}^c \ .(10)$$

Deriving this energy by respect to X_j^i we obtain

$$\frac{\partial E}{\partial X_j^i} = \sum_{m=0}^{N(\bar{X}_j)-1} 2\left(X^{c_m(\bar{X}_j),i} - F^{c_m(\bar{X}_j),i}\right.$$

$$\left. - l^{c_m(\bar{X}_j),i} \sum_{k=0}^{2} l^{c_m(\bar{X}_j),k} \left(X^{c_m(\bar{X}_j),k} - F^{c_m(\bar{X}_j),k}\right)\right)$$

$$+ \alpha \sum_{m=0}^{N(\bar{X}_j)-1} \sum_{n=0}^{N_t(c_m(\bar{X}_j))-1} \sum_{l,l'=0,j(l')=j}^{N_p(c_m(\bar{X}_j))-1} 2X_{j(l)}^{c,i} \int_{K_n^c} (\nabla\phi_l^c)^T D(u^c, v^c)\nabla\phi_{l'}^c \ (11)$$

for every $i = 0, 1, 2$. In order to find the minimum of this energy, these equations are equaled to 0. A system of $3N_p$ equations and unknowns is thus generated.

4.1   Solving the System

We are going to see how to build up the system and solve it numerically. We express the system as $Au = b$ where A is a square matrix, b is the independent constant vector and u is the vector of $3N_p$ unknowns to be ordered in the following way

$$u = (X_0^0, X_1^0, ..., X_{N_p-1}^0, X_0^1, ..., X_{N_p-1}^1, X_0^2, ..., X_{N_p-1}^2) \ . \tag{12}$$

We will concentrate on (11) to derive both matrix A and vector b. To compute vector b we realize that constant values generated from (11) are only affected by the first term in the following manner

$$b_{j+iN_p} = \sum_{m=0}^{N(\bar{X}_j)-1} \left(F^{c_m(\bar{X}_j),i} - l^{c_m(\bar{X}_j),i}\left(\sum_{k=0}^{2} l^{c_m(\bar{X}_j),k} F^{c_m(\bar{X}_j),k}\right)\right) \ . \tag{13}$$

Both terms of (11) supply information for matrix $A = (a_{j,j'})$. The first term of the energy supply the following information to the diagonal part of matrix A. For every $j = 0, ..., N_p - 1$ and $i = 0, 1, 2$ we have

$$a_{j+iN_p,j+iN_p} + = \sum_{m=0}^{N(\bar{X}_j)-1} \left(1 - \left(l^{c_m(\bar{X}_j),i} \right)^2 \right) . \tag{14}$$

Note that $+=$ in this formulation means accumulation of values. Matrix A is initialized to null values. In the process of constructing this matrix, values are accumulated in different steps to facilitate the task of dividing the algorithm. Later on we will see that the diagonal part of this matrix is affected by the second term of the energy.

For the non-diagonal part of A such a term provides the following:

$$a_{j+N_p,j} = a_{j,j+N_p} + = - \sum_{m=0}^{N(\bar{X}_j)-1} l^{c_m(\bar{X}_j),0} l^{c_m(\bar{X}_j),1}$$

$$a_{j+2N_p,j} = a_{j,j+2N_p} + = - \sum_{m=0}^{N(\bar{X}_j)-1} l^{c_m(\bar{X}_j),0} l^{c_m(\bar{X}_j),2}$$

$$a_{j+2N_p,j+N_p} = a_{j+N_p,j+2N_p} + = - \sum_{m=0}^{N(\bar{X}_j)-1} l^{c_m(\bar{X}_j),1} l^{c_m(\bar{X}_j),2} . \tag{15}$$

If we now concentrate on the second term of the energy we realize that the contribution to matrix A is the same for the 3 coordinates $i = 0, 1, 2$. We examine every camera c and for every camera we go through every triangle K_n^c. Each triangle is composed of 3 vertexes $(x_c(\bar{X}_j), y_c(\bar{X}_j)), (x_c(\bar{X}_{j'}), y_c(\bar{X}_{j'})), (x_c(\bar{X}_{j''}), y_c(\bar{X}_{j''}))$. From every triangle we compute the following contributions to the matrix:

$$a_{j,j} + = \int_{K_n^c} \left(\nabla \phi_j^c \right)^T D(u^c, v^c) \nabla \phi_j^c$$

$$a_{j',j'} + = \int_{K_n^c} \left(\nabla \phi_{j'}^c \right)^T D(u^c, v^c) \nabla \phi_{j'}^c$$

$$a_{j'',j''} + = \int_{K_n^c} \left(\nabla \phi_{j''}^c \right)^T D(u^c, v^c) \nabla \phi_{j''}^c$$

$$a_{j',j} = a_{j,j'} + = \int_{K_n^c} \left(\nabla \phi_j^c \right)^T D(u^c, v^c) \nabla \phi_{j'}^c$$

$$a_{j'',j} = a_{j,j''} + = \int_{K_n^c} \left(\nabla \phi_j^c \right)^T D(u^c, v^c) \nabla \phi_{j''}^c$$

$$a_{j'',j'} = a_{j',j''} + = \int_{K_n^c} \left(\nabla \phi_{j'}^c \right)^T D(u^c, v^c) \nabla \phi_{j''}^c \tag{16}$$

these contributions are the same for the 3 coordinates.

Fig. 2. Stereoscopic pair for Herve's face

4.2   Numerical Scheme

If we study matrix A we will realize that most of its values are null, therefore solving the previous system will induce a lot of unnecessary calculations and, therefore, a very slow method.

We have implemented an efficient numerical scheme for the resolution of the previous system. Studying the way the system matrix A is constructed we realize that it is a sparse matrix and most of its values are situated in symmetrical positions. Also we realize that in most cases, a single point will be surrounded by a small number of triangles, inducing, for every point on the mesh, a small number of values for the second term of the energy.

To take advantage of this we express the system $Au = b$ as $\alpha(D+H)u+Cu = b$ where C is a matrix with the contribution of the first term of the energy, D is the diagonal matrix with the contribution of the second term of the energy and H is the contribution of the non-diagonal part of the second term.

To solve this system we use a fixed point equation:

$$u = (\alpha D + C)^{-1}(b - \alpha Hu) \; . \tag{17}$$

5   Experimental Results

In this section we present some experiments. For the first experiment we propose a single stereo pair (Hervé's face). The second is a bust for which we have a complete round sequence of 47 calibrated images.

5.1   Single Stereo Pair

In Fig. (2) he have a stereo pair of a face. The purpose of this experiment is to show the validity of the method when there is only a single disparity map estimated for each camera. In Fig. (3) we show the profile and front of the Herve's reconstruction obtained through the method explained in [1]. We have only reconstructed the region belonging to the face.

During the process we have to construct two triangulations to be used within the finite element method. In Fig. (4) we show the Delaunay triangulations for

Fig. 3. Profile and front views of the Herve's 3D reconstruction
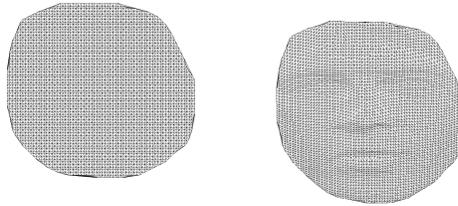
Fig. 4. Camera triangulations for the Herve's stereoscopic pair

both cameras. The reference system is situated on the left camera. We may appreciate that for the left image we have a regular mesh. This is because the points in the triangulation are coincident with the pixel distribution on the image. For the right image the points are a little warped in correspondence with the displacement given by the disparity map. The shape of the face could be guessed there. In Fig. (5) we have several 3D regularizations using different values for α and s. The values of α increase by rows and the values of s increase by columns. As we can see the models become more regular for the last rows and more isotropically regularized for the last columns.

From these examples it is easy to see that s plays the role of an anisotropic/isotropic diffusion parameter as it does for other image diffusion related applications. The bigger s is the more the image is isotropically regularized. Concentrating on the first row: The first image on the left is computed for $\alpha = 0.1$ and $s = 0.1$ so the model is little regularized and its geometry (given by the disparity map) is preserved. On the other hand (third row and the right image) $\alpha = 1.0$ and $s = 1.0$ the model is regularized in all directions and the structures are also smoothed.

5.2   Stereo Sequence

Now we consider a more complex sequence. In this case the sequence is composed of 47 images taken around a face bust. Figure (6) shows the configuration of this
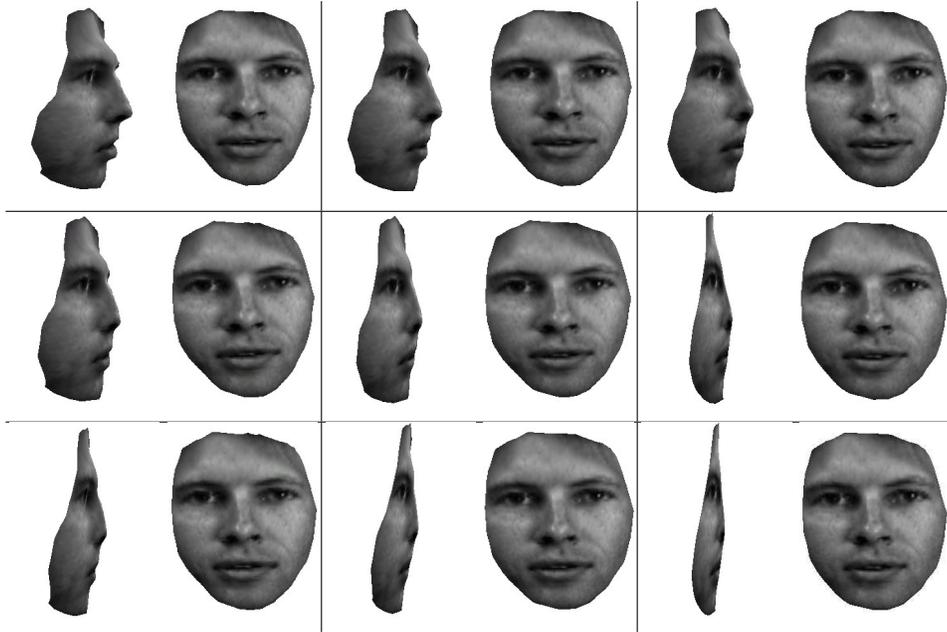
Fig. 5. Profile and front views of different 3D regularizations for Herve's sequence. First row: Left pair of images, $\alpha = 0.1$ and $s = 0.1$; center images, $\alpha = 0.1$ and $s = 0.5$ and; right images, $\alpha = 0.1$ and $s = 1.0$. Second row: Left images, $\alpha = 0.5$ and $s = 0.1$; center images, $\alpha = 0.5$ and $s = 0.5$; and right images, $\alpha = 0.5$ and $s = 1.0$. Third row: Left images, $\alpha = 1.0$ and $s = 0.1$; center images, $\alpha = 1.0$ and $s = 0.5$; and right images, $\alpha = 1.0$ and $s = 1.0$

sequence with the projection planes of the cameras. This is a closed sequence in where the first and last images are correlatives.

In Fig. (7) the 3D reconstruction for the bust sequence using the technique explained in [1] is shown. In Fig. (8) we show the triangulation and image for camera 6 for some set of 3D points. In this figure we may guess the shape of the faces and the set of triangles inside it.

Figures (9) and (10) contain several regularizations using a range of values for α and s. We may appreciate that when α is increased the set of points becomes more regular and when s tends to 1 the regularization becomes more isotropic.

Some of the experimental results for the Bust sequence may be accessed at the web site `http://serdis.dis.ulpgc.es/~jsanchez/research/demos/`.

6   Conclusions

In this paper we have presented a novel method for the regularization of a set of 3D points. We have established an energy in a traditional attachment–regularizing couple of terms. In the regularizing term we have made use of an

Fig. 6. Bust configuration: This figure shows the 3D reconstructed bust and the distribution of the projection planes corresponding to the 47 cameras

Fig. 7. Original 3D reconstruction for the Bust sequence

operator similar to the Nagel–Enkelmann operator for 3D regularizations. This new method has been embedded into a 2D finite element approach to take advantage of the underlying precision of data. Then we have managed to derive this energy and propose a very efficient and efficient numerical scheme that allows us to speed up the process and reduce the memory needs.

We have shown in the experiments that varying the α parameter results in a more regular set of points and varying the λ parameter implies a more regular set of points by preserving disparity map discontinuities as we have expected from the results obtained in other fields.

One of the main advantages of the method is that it regularizes sets of unstructured 3D points without using any geometric relation in 3D. In particular, this method could be used as a preprocessing step before the construction of 3D surfaces fitting the 3D points. Most of the techniques for such surface re-
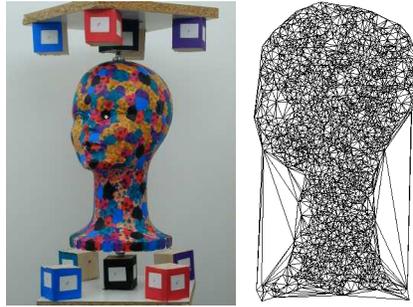
Fig. 8. Triangulation generated by the set of 3D points on the 6th projection plane

construction are very sensitive to the noise in the 3D points representation, and they require the set of 3D points to be regular enough to work properly.

7   Acknowledgments

References

1. L. Alvarez, R. Deriche, J. Sánchez and J. Weickert. Dense disparity map estimation respecting image derivatives: a PDE and scale-space based approach. *Journal of Visual Communication and Image Representation*, 13:3–21, January 2002. Also published as Inria Research Report n° 3874
2. L. Alvarez, J. Weickert and J. Sánchez. Reliable Estimation of Dense Optical Flow Fields with Large Displacements. *International Journal of Computer Vision*, 39(1):41–56, 2000.An extended version maybe be found at Technical Report n°2 del Instituto Universitario de Ciencias y Tecnologías Cibernéticas of the Universidad de Las Palmas de Gran Canaria, Spain
3. N. Amenta and M. Bern. Surface Reconstruction by Voronoi Filtering. *14th ACM Symposium on computational Geometry*, pages 248–253, June 1998
4. F. Bernardini, J. Mittleman and H. Rushmeier and C. Silva. The Ball Pivoting Algorithm for Surfarce Reconstruction. *IEEE Transactions on Visualization and Computer Graphics*, 5(4):349–359, Oct.–Dec. 1999
5. H. Edelsbrunner and E. P. Mucke. Three-dimensional Alpha Shapes. *ACM Transactions on Graphics*, 13(1):43–72, January 1994
6. O. Faugeras. Three-Dimensional Computer Vision: A Geometric Viewpoint. MIT Press, 1993
7. O. Faugeras and R. Keriven. Complete Dense Stereovision Using Level Set Methods. *Proceedings of Fifth European Conference on Computer Vision*, 1998
8. O. Faugeras and R. Keriven. Variational principles, surface evolution, PDEs, level set methods and the stereo problem. *IEEE Transactions on Image Processing. Special issue on Geometry driven diffusion and PDEs in image processing*, 7(3):336–344, 1998

Fig. 9. Profile and front views of two different 3D regularizations for the Bust sequence. First row: Two views for $\alpha = 0.1$ and $s = 0.01$. Second row: Two views for $\alpha = 2.0$ and $s = 0.01$

9. O. Faugeras, and Q. Luong and T. Papadopoulo. The Geometry of Multiple Images. Mit Press, 2001
10. P. George and H. Borouchaki. Triangulation de Delaunay et Maillage. Hermes, 1997
11. R. Hartley and A. Zisserman. Multiple View Geometry in Computer Vision. Cambridge University Press, 2000
12. H.H. Nagel and W. Enkelmann. An Investigation of Smoothness Constraints for the Estimation of Displacement Vector Fields from Images Sequences. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(5):565–593, 1986

Fig. 10. Profile and front views of two different 3D regularizations for the Bust sequence. First row: Two views for $\alpha = 16.0$ and $s = 0.01$. Second row: Two views for $\alpha = 16.0$ and $s = 1.0$