Template Updating Based on Context

4 authors:

Cayetano Guerra Artal
Universidad de Las Palmas de Gran Canaria
42 PUBLICATIONS **387** CITATIONS
SEE PROFILE

Josep Isern
Universidad de Las Palmas de Gran Canaria
52 PUBLICATIONS **274** CITATIONS
SEE PROFILE

Antonio Carlos Domínguez Brito
Universidad de Las Palmas de Gran Canaria
63 PUBLICATIONS **339** CITATIONS
SEE PROFILE

Daniel Hernández-Sosa
Universidad de Las Palmas de Gran Canaria
92 PUBLICATIONS **645** CITATIONS
SEE PROFILE

Template Updating Based on Context

Cayetano Guerra Artal
IUSIANI
Univ. Las Palmas de G.C.
cguerra@iusiani.ulpgc.es

Josep Isern González
IUSIANI
Univ. Las Palmas de G.C.
jisern@dis.ulpgc.es

Antonio Domínguez Brito
IUSIANI
Univ. Las Palmas de G.C.
adominguez@iusiani.ulpgc.es

Daniel Hernández Sosa
IUSIANI
Univ. Las Palmas de G.C.
dhernandez@dis.ulpgc.es

Abstract

Visual tracking based on pattern matching is a very used computer vision technique in a wide range of applications [3]. Updating the template of reference is a crucial aspect for a correct working of this kind of algorithms. This work proposes a new approach to the updating problem in order to achieve a better performance and robustness of tracking. The proposed technique has been compared experimentally with other existing approaches with excellent results. The most important improvements of this approach is its parameter-free working, therefore no parameters have to be set up manually in order to tune the process. Besides, objects to be tracked can be rigid or deformable, the system is adapted automatic and robustly to any situation. An added memory of patterns allows to improve the robustness.

1. Introduction

Visual tracking based on pattern matching is a very used computer vision technique in a wide range of applications [3]. Its working is simple, a template of reference is searched in the current image. Due to that the view of the tracked object can modify its aspect through the time, an update of the template is necessary in order to keep the object.

Two problems can arise because of the insufficient or excessive frequency of the number of updates. In the first of them, the visual aspect of the object of interest can become too different from the pattern and, in this way, the searching algorithm can find other part of the searching window more similar to the current pattern. This produces a *jump* in the object of interest. The other problem is due to applying too and unnecessary updates to the template of reference. The digital nature of images and patterns can cause *drifting* due to an accumulative sub-pixel error in every update. Sometimes, the random movement of the object can counteract the effect of the drift, but generally the drift can be significant and end up losing the object. Every update causes a potential drift.

This work proposes a new template updating approach within the framework of representation spaces based on second order isomorphisms. Among its advantages are a parameter free working, no parameters have to be set up manually prior working, and a better performance than the traditional updating methods.

2. Updating the template

A number of strategies have been proposed to define the template to be used during

Actas del Simposio de Reconocimiento de Formas y Análisis de Imágenes (AERFAI), pp.159-166
ISBN: 84-9732-445-5

the tracking process. In [4, 5] there are good surveys about these techniques. Strategies go from no template updating at all, others with very naive approaches and some of them using similarity thresholds. Among them, *template update based on statistics* [6] tries to overcome the inherent problems of drifting and jumps of interest seeking a balance in the number of updates to perform. This updating schema takes into account that exceeding a similarity threshold provides only local information and does not provide information at all about the rest of the image. However, the quality of a maximum (or minimum) relies on the values that surround it. Therefore, this statistical method of updating considers the rest of values of the similarity function, in such a way that if the maximum (or minimum) is differentiated enough from the rest of values then the quality of the current pattern is good. This level of differentiation is calculated based on a statistical function.

None of these mentioned methods offer a reliable solution to the problem of jumps in the object of interest. The following sections will describe the proposed updating template algorithm.

2.1. Second-Order Isomorphisms

The .ºbjects.ªre located in the real world and, according to Shepard [1], we will name to this world *Distal Space*. Every object in this space will have its own representation in an inner space $\mathbf{\Phi}$, named *Proximal Space*. In this work we define *Visual Object* to any physical entity in the real world which has associated its own internal representation in $\mathbf{\Phi}$. The goal of the visual system is to assign to every visual object in the distal space a unique symbol in a proximal space, and thereby to establish an isomorphism between both spaces, [7].

Besides this correspondence, it is even much more useful to establish relations among objects in a distal space and their respective representations in the proximal space. A *second order isomorphism* [1, 8] should accomplish that if similarity between two distal objects A and B is greater than between distal objects B and C, then the distance between their respective representations (A', B' and C') should verify that $d(A', B') < d(B', C')$. Therefore, the representation schema not only stores information about the objects but also information about their relationships.

3. View-based Representation Spaces

View-based approaches have experienced a renewed interest in the computer vision community in the last decade. After Bergen and Adelson [2], the appearance of a visual object in terms of images is described by the plenoptic function. That is, if the plenoptic function V of a visual object is known, then every possible view of that object can be generated. This function depends on a set of parameters $\vec{x}$, like viewing position and lighting conditions, whose variability defines the appearances subspace corresponding to the visual object [9] in the views space.

If time varying parameters $\vec{\rho}(t)$ are included, the plenoptic function $V(\vec{x}, \vec{\rho}(t))$ will be able of dealing with non-rigid visual objects. Unfortunately, finding the plenoptic function corresponding to an object in a certain scene is a very complex problem.

In order to overcome this drawback much effort has been done in the study of the views space. To characterize precisely the variability of images and other perceptual stimuli, a mathematical approach can be taken.

The views space can be modeled in image coordinates, based on considering the set of $n \times m$ pixels corresponding to each image as a $R^{m \times n}$ vector. We can consider each image as a vector with dimension $m \times n$. The set of all possible images of any distal object is a continuous subset of the views space [10]. This continuity is related to the smooth variation of visual aspect with respect to the plenoptic parameters. This can be stated as a *continuity principle* in the following manner: given arbitrarily small τ and δ_d, the following condition will met:

$$d[V(\vec{x}; \vec{\rho}(t)), V(\vec{x}; \vec{\rho}(t+\tau))] \leq \delta_d, \ \forall \vec{x} \in \mathbf{S} \quad (1)$$

Where $\mathbf{S}$ corresponds to the support set of V

and d is a defined distance function. Varying t, in the generalized case, the set of points corresponding to the images of a distal object are in a manifold [10] $\mathcal{M}_x$ of the Views Space. The manifold $\mathcal{M}_{\vec{x}}^{O}$ of a certain object O is a lower dimensional subspace embedded [11, 9] in the views space with the l parameters of the plenoptic function as intrinsic dimensions:

$$\mathcal{M}_{\vec{x}}^{O} = \{V(\vec{x};\vec{\rho}(t)) \mid \vec{\rho} \in R^{l}\} \qquad (2)$$

During a tracking process of an object, this does not show all possible views of itself included in its manifold but just a subset of them. This manifold subset, $I(\vec{x};t) \subset \mathcal{M}_{\vec{x}}^{O}$, will shape as a parametric curve of the time. We name this curve *Visual Transformation Curve of the Object*.

The tracking process tries to follow the visual object through this curve obtaining the values α_0 corresponding to the location of the best match at time t, as the following minimization:

$$\vec{\alpha_0}(t) = arg\min_{\alpha} \sum_{\vec{x}} d(W(I(\vec{x};t);\vec{\alpha}), T(\vec{x};t)) \qquad (3)$$

Where $I(.;.)$ is the image where looking for the template $T(.;.)$ by means of a windowing function $W(.;.)$, which extracts an area of the same size than $T(.;.)$ at position $\vec{\alpha}$. $\vec{\alpha_0}(t)$ will be the minimum of the matching function, valued over all possible values of $\vec{\alpha}$, that is, the position of the window W over the image V.

The template tracking depends on the definition of several elements. Once defined the matching strategy and distance function to be used, the fundamental element to be defined is the template update strategy or, in other words, the steps in which the visual transformation curve is sampled.

4. Proposed Template Matching Updating Technique

In order to describe the procedure proposed in this paper, we will denote as $\vec{p}$ to a point corresponding to the representation of a visual object in the proximal space $\mathbf{\Phi}$ at a certain time, i.e. $\vec{p}$ will correspond with the template

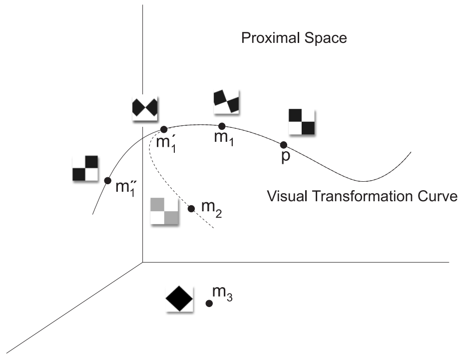

Figura 1: The figure depicts the points corresponding to the symbolic representations of the different searches over an image in a tracking process. The dotted line illustrates the consequences of a lack of required updating.

T (see expression 3) in the space defined by $R^{m\times n}$. A distance d can be established in $\mathbf{\Phi}$. In this work, the distance d between two points $\vec{p_1}$ and $\vec{p_2}$ is based on the L_2 norm. This distance will be used between input image and template in order to obtain the best match.

After applying the distance function between image and pattern sliding the template over the searching window according to expression 3, a variable number of local minima will show up, among them, the absolute one. In $\mathbf{\Phi}$, see figure 1, the vector $\vec{p}$ corresponds to the pattern of reference, i. e. the view of the object of interest to look for. The vector $\vec{m}_1$ will be the absolute minimum since it is the visual object most similar to the object of interest. The existence of more local minima, $\vec{m}_2$ and $\vec{m}_3$, implies that there are other similar objects in a certain degree to the object of interest. We name them *objects of the context.* These objects, like the object of interest, have also their own curve of visual transformation included in their manifolds of the proximal space. Although for the sake of simplicity these objects of the context will remain static. The *Visual Transformation Curve* of the object of interest is the loci of the points corresponding to the different minima after the matching process on input images during a certain time. This curve will be composed by the nearest vectors $(\vec{m}_1, \vec{m}_1', \vec{m}_1'', ...)$ to the pattern of reference $\vec{p}$. Therefore, $\vec{m}_1$ corresponds to the closest point to $\vec{p}$ in the moment $t = 0$, $\vec{m}_1'$ corresponds to the closest point to $\vec{p}$ in the time $t = 1$ and so on. However, if there exist, at least, one object of the context, $\vec{m}_2$, and the pattern of reference $\vec{p}$ is not updated, it may occur that, after a number of frames, the absolute minimum does not correspond to the real object of interest but to the most similar object of the context, as figure 1 shows. Thus, the area of the searching window corresponding to the point $\vec{m}_2$ will be taken as the object of interest, resulting in an error of the tracking process, that is an *interest jump error*, which is a very common error of updating techniques that do not update the pattern just in time.

The origin of the problem is caused by the lack of updating or an inappropriate updating rate of the pattern of reference. It can be seen in figure 1, c) that $d(\vec{p}, \vec{m}_1) < d(\vec{p}, \vec{m}_2)$ and $d(\vec{p}, \vec{m}_1') < d(\vec{p}, \vec{m}_2)$ but $d(\vec{p}, \vec{m}_1'') > d(\vec{p}, \vec{m}_2)$. For the sake of clarity the most similar object of the context, $\vec{m}_2$, does not move and consequently does not draw any visual transformation curve.

It is clear that the pattern should be updated before any object of the context can be more similar to the pattern of reference than the current view of the object of interest. To accomplish this an updating threshold must be set up taking into account the closeness of the *objects of the context*. Therefore, when a new view of the object of interest is taken as current pattern a new updating threshold is also computed automatically. The assigned value can be obtained by the rule of dividing by two the distance to the closest object of the context to the new pattern.

5. Realigning the template

Drifting is always possible as long as updating takes place, therefore realignment is recommended. Some works have been done recently to deal with this matter [4]. A used way for realigning the current template is by taking the first or previous templates with some modifications.

In real situations many objects have a cyclic behavior showing views repeatedly of itself to an observer. For example, in figure 4 can be seen how a person moves his head on both sides offering different views of his face again and again.

This work proposes a realignment of the template based on previous views of the object of interest corresponding to templates selected by the context based updating algorithm as current templates. Therefore, the different current templates are always significant views of the object of interest in function of its context. A bank of templates is used to store views of the object of interest acting as a short term memory.

A heuristic meaning drives this process on the basis that previous templates will have greater probabilities than the current one of being correctly aligned. Experimentally, this conception has given good results.

This proposal has further consideration since the process of views extraction can be understood as a manifold sampling of an object in its proximal representation according to a distinguishable criteria in its context. These views will be representative enough in relation to its visual context, instead of simple constant sampling criteria. (perceptual grounding)

5.1. Renewing the bank

Once the bank of patterns is full, it is required a strategy in order to extract and insert the appropriate patterns. It must be taken into account an algorithm that assigns an *index of utility* to every pattern in the bank. The patterns with less index, i.e. a pattern with the less probability to be used, are replaced by new ones. The index of utility proposed is calculated with a formula that uses the concepts of *persistence* and *obsolescence*. Persistence and obsolescence in every pattern are updated every cycle. Therefore, every pattern in the bank owns an utility index that adjusts its relevance or importance. This index is defined as:

$$U(t,i) = \frac{t_p(i)}{1+t-t_s(i)} \quad (4)$$

In (4) U corresponds to the index of utility for an index pattern in the bank i and a time (or frame) t, t_p and t_s correspond to persistence and obsolescence respectively.

We name *persistence* of a pattern to the amount of time or number of consecutive frames that this pattern is taken as current pattern, i.e. the continuous interval of time or working cycles that a pattern has been used to carry out the tracking. Persistence shows clearly when a certain pattern belonging to the object of interest is a common view of it.

Obsolescence of a pattern is defined as the time elapsed since the last time it was taken as current pattern. It is assumed that a pattern with high obsolescence implies a low probability of that pattern to be reused.

6. Experiments

Pattern updating is necessary if the view of the object of interest changes through the time. Besides, this updating must be done at the right moment in order to avoid the two most significant errors in a tracking process: *drifts* and *interest jumps*. These two kinds of error will mark experimentally the goodness of the different updating approaches.

Several experiments have been done in order to evaluate the performance of the proposed solution. Among them, two critical sequences, described in this paper, demonstrate the higher level of robustness of the new approach in comparison with the existing updating methods. Actually, only statistics updating based method [6] is used as the other methods are too simple and their limitations are obvious.

A complete tracking module has been developed to carry out the presented experiments. To obtain the results only the updating schema of this algorithm has been changed. In order to evaluate the updating approaches, the best method will be the one that carries out a correct tracking (without interest jump nor drifting errors) with the smallest number of updates.

In the first experimental sequence, see figure 2, a person walks and her face is tracked. At first sight, it seems a not problematic task. However, an error happens due to the existence of local minimums near the absolute one, and all of them surrounded by a very different environment. Such a situation drives to a not pattern updating, and a consequent interest jump error, when the constant threshold and statistic based update algorithms are used.

Things that we perceive or think as quite different may not result be so to a certain similarity function. Figure 2 illustrates such error. Every frame is shown beside its corresponding similarity function. To fix the problem, using the statistic based update method, it is necessary to increase the level of certainty and so the number of updates. Table 1 shows the resulting values of the two compared algorithms. Carrying out both of them a correct tracking process the difference raises in the number of updates needed. The less number of updates the less probability of drifting. The second experiment shows how the proposed updating method can adapt the rate of updates according to the proximity of very similar objects. In figure 3 can be seen a frame of a four seconds sequence where the object of interest is a fish that swims into a shoal, so it is surrounded

Statistical approach

Reliability threshold	Number of updates	Jump errors
0,75	-	Yes
0,80	-	Yes
0,85	-	Yes
0,90	468	No

Context based approach

Reliability threshold	Number of updates	Jump errors
–	106	No

Cuadro 1: Above, number of updates based on statistical pattern updating and errors of jumps for different reliability threshold. Below, number of updates in the same sequence using context based pattern updating approach

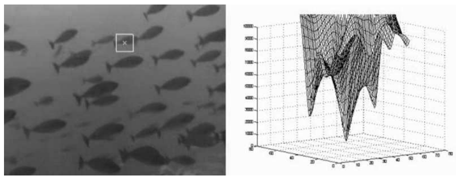

Figura 3: The figure shows a frame of a sequence where the object of interest is surrounded by very similar others. The right function depicts the shape of the scene in the representation space. The absolute minimum corresponds to the object of interest and the local minima are objects of the context.

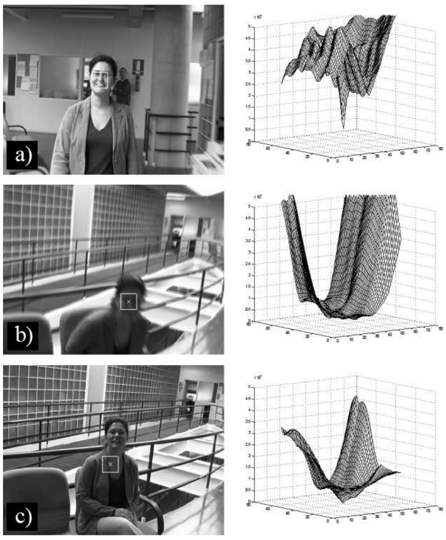

Figura 2: Different frames of a sequence where a face is tracked. Sometimes, at first sight and if similarity function is not displayed, it does not look that the minimum of the similarity function can be so confused.

by other very similar fishes. In order to avoid the loss of the object, the frequency of updates should be high due to the current pattern can be rapidly confused with objects of the context. The function next to the frame shows graphically the object of interest, as the absolute minimum, and the objects of the context (two fishes) as local minima nearest to the absolute minimum.

Statistical approach

Reliability threshold	Number of updates	Jump errors
0,75	-	Yes
0,80	-	Yes
0,85	28	No

Context based approach

Reliability threshold	Number of updates	Jump errors
–	26	No

Cuadro 2: Number of updates needed by the two methods in order to achieve a correct tracking.

6.1. Experimental results adding a bank of patterns

Several experiments have been carried out over a number of sequences with certain grade of repetitiveness in the behavior of its object

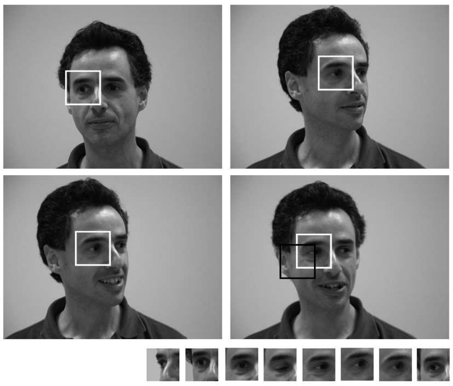

Figura 4: The figure shows the process of tracking over one eye. The white squares depict the tracking using template bank based realignment. The black square shows the result due to the drifting when the bank is not used. The eight patterns below are the templates stored in the bank of patterns.

of interest. All of them show a qualitative common trend, the bigger the size of the bank the smaller number of updates, a bigger number of recoveries and a smaller drifting risk. One of these sequences is shown in figure 4. The figure shows four frames and the object of interest, the right eye, is pointed out with a black rectangle. The table 3 contains in each row the number of templates in the bank with their corresponding number of updates and recoveries obtained. Table 3 shows how the number of updates decreases as long as the amount of previous templates in the bank increases.

Templates #	Updates #	Recoveries #	Drifting
1	80	-	Yes
4	57	80	No
8	53	101	No

Cuadro 3: Number of updating and recovering with an incremental bank size

7. Summary and Conclusions

As conclusions from the experiments carried out in a wide range of environments and conditions we can state three major ones:

- Minimization of the required number of updates achieving a correct tracking process, and minimizing the drift risk.
- Achievement of an automatic template updating method for any environmental condition.
- Improvement the accurateness of the tracking process reducing the drift error by using a bank of patterns.
- Obtaining a computationally light update algorithm in low cost general purpose computers.

Referencias

[1] R. N. Shepard and S. Chipman, "Second-Order Isomorphism of Internal Representations: Shapes of States" *Cognitive Psychology*, Vol. 1, pp. 1-17, 1970.

[2] E.H. Adelson and J.R. Bergen, "The Plenoptic Function and the Elements of Early Vision" *Computational Models of Visual Processing, pp.3-20, Cambridge, MA: MIT Press, 1991*

[3] L. G. Brown, "A Survey of Image Registration Techniques" ACM Computing Surveys*, Vol. 24, pp. 325-376, 1992*

[4] I. Matthews and T. Ishikawa and S. Baker, "The Template Update Problem" IEEE IEEE Trans. Pattern Anal. Mach. Intell.*, Vol. 26, N. 6, pp. 810-815, 2004*

[5] M. Andersson, "Tracking Methods in Computer Vision", Master Thesis. Computational Vision and Active Perception Laboratory (CVAP), 1994.

[6] H. Inoue and T. Tachikawa and M. Inaba, "Robot Vision System with a Correlation Chip for Real-time Tracking, Optical Flow and Depth Map Generation" IEEE International Conference on Robotics and Automation*, pp. 1621-1629, May, 1992.*

[7] S. Edelman, "Representation is Representation of Similarities" Behavioral and Brain Sciences*, Vol. 21, pp. 449-498, 1998.*

[8] *S. Edelman, "Representation and Recognition in Vision"* The MIT Press, *1999.*

[9] *P. Belhumeur and D. Kriegman, "What Is the Set of Images of an Object Under All Possible Illumination Conditions?"* Int. Journal of Computer Vision*, Vol. 3, Num. 28, pp. 245-260, 1998.*

[10] *H. Sebastian Seung and Daniel D. Lee, "COGNITION: The Manifold Ways of Perception"* Science*, Vol. 290, pp. 2268-2269, 2000*

[11] *V. Colin de Verdière and J. L. Crowley, "Visual Recognition Using Local Appearance"* ECCV*, pp. 640-654, 1998.*