International Journal of the Physical Sciences Vol. 6(12), pp. 2799-2806, 18 June, 2011
Available online at http://www.academicjournals.org/IJPS
DOI: 10.5897/IJPS11.420
ISSN 1992 - 1950 ©2011 Academic Journals

Full Length Research Paper

Video motion perception using optimized Gabor filter

Elarbi-Boudihir, M., Rehman, A.* and Saba, T.

AI and Robotics Research laboratory, Computer Science Department, Imam Muhammad Bin Saud University, Riyadh, Kingdom of Saudi Arabia (KSA).

Accepted 17 May, 2011

Motion estimation in image sequences is a fundamental issue in many applications as for instance in artificial vision and three-dimensional scene reconstruction. The basic problem of the recent spatio-temporal filtering techniques used in the perception of object motion from a time sequence of 2-D images is the computational burden. This is mainly due to the computation of the frequency responses of the images to the Gabor filters using the multi-resolution approach. In this paper we present a method for detecting and estimating object motion using spatio-temporal frequency information from image sequence. Our model of visual perception of object motion from a video stream is based on a contributory adaptation of spatiotemporal Gabor filters. This method uses a bank of Gabor filters which are frequency tuned and limited in spatial extent. Instead of testing the entire filter bank to determine the appropriate filter parameters, a genetic algorithm is used to derive the filter subset that provides the object texture information and optimize the search for its perceived motion. This model has been evaluated on artificial as well as natural image sequences. The results obtained show the feasibility of the approach which attains a reasonably fast output rate (several images/second) for a better resolution.

Key words: Dynamic vision, motion estimation, segmentation, genetic algorithms, Gabor filters.

INTRODUCTION

Modeling visual perception of object motion in a video sequence offers various areas of research for the development of real-time models of dynamic perception-action. Most of the vision systems extract features to build essential blocks for recognition and high level scene interpretations. Most of the time, some of the low quality features lead to wrong interpretations. This conclusion is noticed whenever object motion detection and estimation is concerned (Cornea et al., 2007; Dorfmüller-Ulhaas, 2003). It is generally due to the fact that the computer should interpret a huge amount of data from an image sequence by using algorithms whose design is driven by a limited time interval. Consequently, a need for an optimization stage for systems relying on image data is becoming necessary. Such systems include, for example, target tracking systems, autonomous vehicles and vision-based industrial robots. For those systems, a correct interpretation is tightly linked to the accurate detection and estimation of image motion as described by (Verri and Poggio, 1988). For any vision system the image motion information available is simply the projection of 3-D real velocity on the image plane (Prazdny, 1983). Therefore, motion analysis is confined to estimate the apparent motion and then to evaluate the relevant 3-D motion. It should be noticed that most, if not all, techniques proposed in the literature for 3-D motion estimation process take for granted the estimation of a motion field (Maunsell and Ferrera, 1995; Macenko et al., 2007; Goldberg, 1989). The resultant simplification tends to imply that the 2-D motion field is an error free process, which is unfortunately false. The ill-posed problem in motion detection and estimation is heavily affected by the discontinuity of the motion field due to the presence of noise, the presence of occlusions and finally the aperture problem (Elarbi-Boudihir, 2008).

In the literature, two basic strategies are identified for detecting object motion in image sequence: matching techniques (correspondence), and gradient-based approaches which can be classified into regularization and multiconstraint-based techniques. The gradient methods depend on estimating first and second-order derivatives of pixel intensity. This makes them less immune to image noise and exposed to a high error rate. The correspondence methods are simple to implement

*Corresponding author. E-mail: amjadbzu2003@yahoo.com

but can easily degenerate with combinatorial explosions of matching possibilities, especially in complex scenes (Khelifi et al., 2004).

The methods described previously consider the optical flow estimation as an ill-posed problem as many other inverse problems in computer vision according to Hadamard theory. When the optical flow approach is used, two other strategies may be used: spatio-temporal filtering, and phase-based methods. The latter is a hybrid method where the motion field estimates are obtained by using differential methods applied to the phase instead of the image intensity. This necessitates a very large bank of filters and a large number of images in the sequence. Consequently, the phase-based method is relatively accurate, but stresses any hardware and software component of current computing systems, and finally performs best in single object scenes only. On the other hand, spatio-temporal methods perform with a considerable accuracy for most existing off-line vision applications, but still rely on special hardware to supply the computational power needed. Its broad strategy is to avoid the aperture problem at edges just as human visual system does, while still relying on low-level pixel operations. This makes it relatively the best candidate for implementation on advanced real-time vision systems (Maunsell and Ferrera, 1995; Movellan, 2005; Petkov and Subramanian, 2007; Prazdny, 1983). The basic problem resides in the computational burden of computing the frequency responses of the images to the Gabor filters. The classical method of computing the Gabor energy filter response is an $O(kn^2m)$ algorithm where k is the size of the convolution kernel, m is the number of images in the sequence, and each image has n pixels. In fact, the speed of the spatio-temporal method may be reduced if the set of the filters are properly designed, and their number effectively optimized. In this paper, a genetic algorithm is used to determine a filter subset from the entire Gabor filter bank appropriate for the segmentation and classification process. This filter subset is used to track moving objects and estimate their displacement from image sequences. Moreover, texture detection and orientation estimation is found to be a pre-requisite for any method that uses spatio-temporal frequency information from image sequences. Except for the spatio-temporal filtering method used in this paper, motion analysis does not necessarily use this type of information. In this paper, the spatio-temporal filtering approach is shown subsequently, after which the method used to estimate the optical flow is given. Furthermore, the study illustrates the implementation of the genetic algorithm search optimization before it finally shows the experimental results.

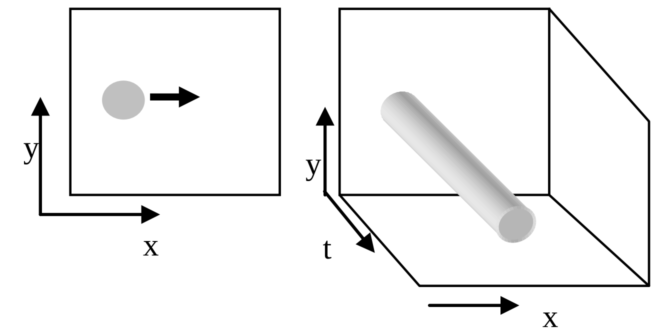

Figure 1. Space-time object movement.

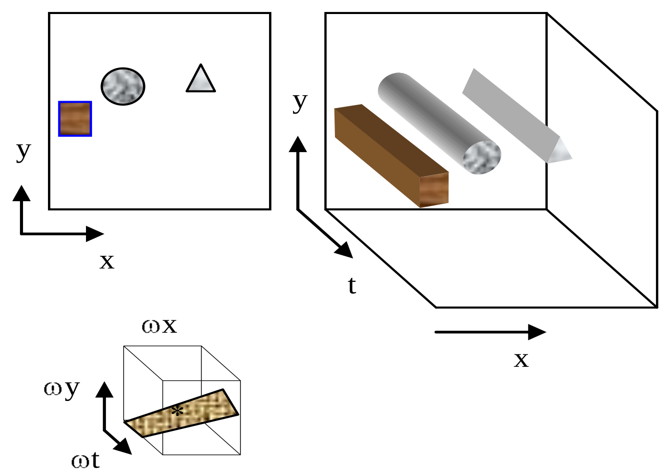

Figure 2. Four Different textures correspond to four points in the space-time frequency domain.

SPATIO-TEMPORAL FILTERING

The recent trends toward developing spatio-temporal filtering schemes in computer vision is justified by the link that they do have in common with biological vision systems. Many researchers in the field claim that the vision zones of the human brain act as spatio-temporal band-pass filters (Petkov and Subramanian, 2007). Nevertheless, concerning motion detection, animal vision systems remain highly superior to the actual artificial vision systems. Motivation stems from the observation that for objects that typically move, recognition is often far easier when viewed in motion rather than in a single static snap-shot (Kamarainen et al., 2006). Such objects may exhibit characteristic temporal signatures which provide additional clues for recognition.

The texture segmentation and temporal tracking methods used rely on temporal coherences between successive images of a sequence (Wang et al., 2006). The approach is to consider the time sequence of 2-D images as 3-D a spatio-temporal image as illustrated in Figure 1.

Then, 3-D filters are applied to the volume image in space-time. Consequently, if the filters are chosen to be oriented, localized and sensitive to a graded range of spatio-temporal frequencies, then robust motion detection will be possible. Therefore, the motion detection problem may simply be formulated as finding the 3-D oriented textures in the spatio-temporal volume indicated by Figure 2.

As indicated earlier, the spatio-temporal filtering method strategy is to avoid the aperture problem while relying on low-level pixel operations. This makes it easier for implementation on advanced real-time systems using dedicated video-chips.

It is easily seen that the spatial object movement may be identified with the orientation attribute in the space-time frame. This means that if a vertical bar moves horizontally in the x-direction of the xy-plane, it generates in the x-y-t volume a solid which orientation may be easily detected using a 3-D edge detector. Let's suppose a textured 2-D object having spatial frequencies (ω_x, ω_y) in the x and y axes respectively. When the object moves with a velocity (*u,v*), its temporal frequency will be given by:

$$u\omega_x + v\omega_y = -\omega_t \quad (1)$$

For best illustrating this, the translation of an image f(x,y) with a velocity *(u,v)* generates a spatio-temporal volume defined by:

$$f(x,y,t) = f(x,y) * \delta(x-ut, y-vt) \quad (2)$$

where * represents a convolution and δ the 2-D delta function. The Fourier transform of Equation (2) gives:

$$\begin{aligned}
\hat{f}(\omega_x,\omega_y,\omega_t) &= \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} f(x,y,t)e^{-j(x\omega_x+y\omega_y+t\omega_t)}dxdydt \\
&= \int_{-\infty}^{+\infty} e^{-jt\omega_t}\int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} f(x,y)*\delta(x-ut,y-vt)e^{-j(x\omega_x+y\omega_y)}dxdydt \\
&= \hat{f}(\omega_x,\omega_y)\int_{-\infty}^{+\infty} e^{-jt\omega_t}\int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} \delta(x-ut,y-vt)e^{-j(x\omega_x+y\omega_y)}dxdydt \\
&= \hat{f}(\omega_x,\omega_y)\int_{-\infty}^{+\infty} e^{-jt\omega_t}\int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} e^{-j(x\omega_x+y\omega_y)}dxdydt \\
&= \hat{f}(\omega_x,\omega_y)\delta(t\omega_t)\delta(u\omega_x+v\omega_y) \\
&= \hat{f}(\omega_x,\omega_y)\delta(t\omega_t+u\omega_x+v\omega_y) \quad (3)
\end{aligned}$$

The last expression describes the plane (1) in the space (ω_x,ω_y,ω_t), since:

$\hat{f}(\omega_x,\omega_y,\omega_t)$ is different from zero only when $\delta(u\omega_x, v\omega_y, t\omega_t) \neq 0$

Hence, if an image includes several oriented textures, the ordered triplet ($\omega_x\omega_y\omega_t$) of the solid in the frequency domain belonging to that object forms a plane.

To detect the 3-D object edges, Gabor filter gives a very efficient result since the orientation of this filter may change making it very sensitive to textures (Movellan, 2005). A 3-D spatio-temporal Gabor filter is simply a sinusoid in a gaussian window:

$$Gs(x,y,t) = \frac{1}{\sqrt{2}\pi^{3/2}\sigma_x\sigma_y\sigma_t}\exp\left[-\left(\frac{x^2}{2\sigma_x^2}+\frac{y^2}{2\sigma_y^2}+\frac{t^2}{2\sigma_t^2}\right)\right].\sin(2\pi\omega_{xo}x+2\pi\omega_{yo}y+2\pi\omega_{to}t) \quad (4)$$

The Gaussian window has an extent governed by σ_x, σ_y, and σ_t. Moreover, the triplet (ω_{xo}, ω_{yo}, ω_{to}) gives the central frequency of the filter. The cosine version of the filter *Gc(x,y,t)* is obtained by replacing the sine by the cosine in the expression (4), as illustrated by Figure 3. The filter response is maximum when its central frequency coincides with that of the object texture in motion.

The motion-energy filter is given by the following phase-independent filter that measures the Gabor energy of the spatio-temporal volume:

$$G(x,y,t) = Gs(x,y,t)^2 + Gc(x,y,t)^2.$$

The frequency response of the term *G(x,y,t)* is given by:

$$G(\omega_x,\omega_y,\omega_t) = \frac{1}{4}\exp\left[-4\pi^2\left(\sigma_x^2(\omega_x-\omega_{xo})^2+\sigma_y^2(\omega_y-\omega_{yo})^2+\sigma_t^2(\omega_t-\omega_{to})^2+\right)\right] + \frac{1}{4}\exp\left[-4\pi^2\left(\sigma_x^2(\omega_x+\omega_{xo})^2+\sigma_y^2(\omega_y+\omega_{yo})^2+\sigma_t^2(\omega_{tx}+\omega_{to})^2+\right)\right] \quad (5)$$

Since $G(\omega_x, \omega_y, \omega_t)$ is a gaussian pair, its response will be maximum to any moving object with spatial frequencies (ω_{xo}, ω_{yo}) and a temporal frequency (ω_{to}). In order to avoid contrast effects, several of these filters may be used.

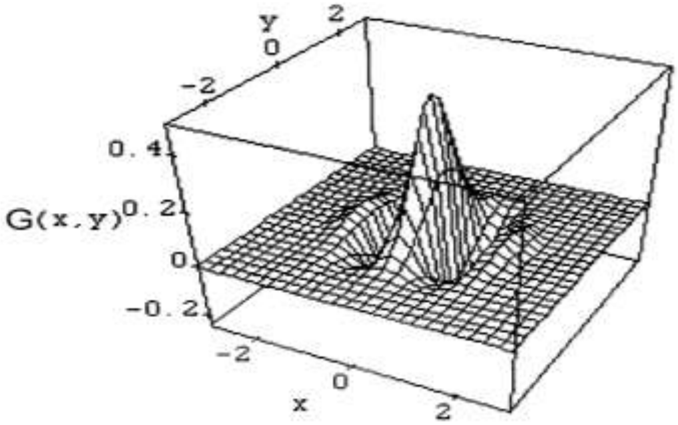

Figure 3. A two-dimensional even Gabor filter (in 3-D, such filters are sensitive to textures).

The main problem with Gabor filtering method is the computational burden of computing the frequency responses of the images to the Gabor filter. It can be noticed that as long as a sequence of images is concerned, this computational burden is not as necessary as it may seem since the number of the selected filters could be reduced to the number of the moving objects in the scene. This selection is done among the Gabor filter bank and it is very important in the detection and estimation process.

OPTICAL FLOW ESTIMATION

Motion detection and estimation are low level cognitive tasks, whereas its segmentation and interpretation are high level cognitive tasks (Horn and Schunck, 1981). Our approach keeps dealing with low level tasks that make use of optical flow estimation. To compute temporal quality, motion information is computed from the reference video sequence in the form of optical flow fields. The same set of Gabor filters used to compute the spatial quality component described previous is used to calculate optical flow from the reference video. The spatio-temporal Gabor decompositions of the reference and test video sequences, and the optical flow field computed from the reference video using the outputs of the Gabor filters can be used to estimate the temporal video quality.

Optic flow methods do not use features, but are nevertheless well-studied. They are based on the image brightness constancy constraint, which assumes that local intensities do not vary significantly from frame to frame. Writing this constancy as a first-order expansion, one obtains an equation linking the optic flow $v = (dx/dy, dy/dt)^T$ and the derivatives of the intensity function:

$$\frac{\partial I}{\partial x}\frac{\partial x}{\partial t}+\frac{\partial I}{\partial y}\frac{\partial y}{\partial t}+\frac{\partial I}{\partial t}=0$$

where $I(x,t) = I(x+\partial x, y+\partial y, t+\partial t)$ is the brightness at point (x, y) of the image.

Assuming that it remains constant over time *t*, the fundamental principles of the various approaches of optical flow estimation are derived: matching (correlation-based), gradient (intensity-based differential), stochastic (relaxation), and spatiotemporal (frequency-based filtering).

Correlation based approaches are based on the local conservation of intensity distribution, which is filtered by a band-pass filter, then processed according to a hierarchical scheme. For that purpose, a matching detection is applied between two consecutive images of a given sequence. This method defines an offset (which is an estimate of the speed) as the best adaptation among the nearby regions that vary in time (Kim et al., 2010). Intensity based differential approaches are based on the intensity conservation of an image within a small interval of time and, as a consequence, they are based on a linear equation with two unknowns. Then, a function of the intensity of an image may be used rather than the image intensity. Finally, the speed may be computed by means of spatio-temporal derivatives of image intensities, assuming that the image domain is continuous (differential) in space and time. Stochastic approaches are based on a constraint that concerns motion discontinuities. They handle occlusions by means of modeling, that is, they model intermittent intensities with the help of stochastic processes or Kalman filters.

The aperture problem is critical with the aforementioned three approaches. We are more interested in the last approach that is described subsequently. Among its advantages, it appears as more robust to the aperture problem.

In frequency-based filtering approaches, the spatio-temporal frequencies are considered along with the speed of motion stimulus, and the optical flow becomes the identification of an energy surface in the space of spatiotemporal frequencies. Movement sensitive mechanisms that are based on the spatially and temporally oriented energy in the space of Fourier have been found to be able to estimate the motion in places where the other approaches fail. Therefore, motion detection in images means here the extraction of spatio-temporal orientations.

In that aim, two sub-approaches are known: phase-based filtering and energy-based filtering. In the first case, the constituents of speed may be defined in terms of immediate movement at phase-level of the outputs of Gabor filters tuned to speed. In the second case, these spatiotemporal oriented filters may be built on separate filters, that is, the spatial and temporal filtering may be sequentially applied to the input image. Obtaining speed would be based on a codification scheme of populations where speed is expressed as the ratio of the outputs of energy filters tuned to various speeds. Assuming linearity, the output of each filter changes with various contrasts but their ratio remains constant. This approach is thus equivalent to correlation-based approaches. The use of spatiotemporal oriented filters that are sensitive to motion stimuli (as Gabor filters) takes its origin in the research about mammal vision, which has revealed the presence of simple cells in V1 and of complex cells in MT that behave as band-pass spatio-temporal filters. The response of these cells depends on the speed of a moving object as well as on its shape and on its frequency characteristics. Among the advantages of energy methods one could mention an intrinsic smoothing which reduces the effects of the "aperture problem"', a better robustness to noise, and good-quality results for natural image sequences. Drawbacks are a high computational cost and a lack of precision near the borders of the movement.

Consequently, using our approach to estimate the object velocity, we should:

(1) Find the oriented textures with frequencies ($\omega_x \omega_y \omega_t$) in the spatio-temporal volume. This may be established using, for instance, three band-pass filters.
(2) Trace the loci of the corresponding points in the spatio-temporal frequency domain
(3) Find the plan which holds the corresponding points
(4) From relation (1), estimate the optical flow (u,v) of the object.

The number of the selected filters could be reduced to the number of the moving objects in the scene. This selection is done among the Gabor filter bank and it is very important in the detection and estimation process. The best way to perform this selection is the use of a genetic algorithm as shown subsequently.

APPLICATION OF GAs

A key issue with this approach is selecting a number of appropriate features. In most cases, relevant features are unknown. Often, a large number of features are extracted to better represent the target; however, without explicitly employing a feature selection strategy, many of them could be either redundant or even irrelevant to the classification task. As a result, classification performance might not be optimum.

Bebis et al. (2002) have shown that it is possible to make two arbitrary patterns similar by encoding them with a sufficiently large number of redundant features. Ideally, we would like to use only features having high separability power while ignoring or paying less attention to the rest. In the context of vehicle detection, it would be desirable if we could exclude features encoding fine details (that is, features that might be present in particular vehicles only). Finding out what features to use in a classification task is referred to as feature selection. A limited yet salient feature set can simplify both the pattern representation and the classifiers that are built on the selected representation (Ramzan et al., 2011).

GAs are part of evolutionary computing (which is an area of the artificial intelligence) and emulate the evolutionary behaviour of biological systems to create subsequent generations that guide the search towards optimal/near-optimal solutions. In a broader usage of the term, a GA is any population-based model that uses selection and recombination operators to generate new sample points in a search space. Each sample point is a chromosome (individual) made of genes (parameters of the problem to solve) and a set of individuals constitutes a population.

Each iteration of a GA involves a competitive selection that eliminates poor solutions. The solutions with high fitness are recombined with other solutions by swapping the parts of a solution with another. Solutions are also mutated by making a small change to a single parameter of the problem. Recombination and mutation are used to generate new solutions that are biased towards regions of the space for which good solutions have already been looked at. GAs differ from more traditional optimization techniques in that they involve a search from a population of solutions (chromosomes), not from a single point. This technique involves generating a random initial population with a given number of chromosomes (made of a set of genes). As shown on the flowchart (Figure 4) the initial population of individuals is created either randomly or by perturbing an input individual. The initialization is not critical as long as the initial population spans a wide range of variable settings (that is, has a diverse population) for Gabor filter samples, through time, the frequency space of an image providing information about oriented, band-pass localized textures. Thus each Gabor filter has a Gaussian profile in the frequency domain which center is at ($\omega_{x0}\omega_{y0}\omega_{t0}$) (Ma et al., 2005). The spatial extent parameters are ($\sigma_x\sigma_y\sigma_t$) and the filter is rotated by a varying angle θ about the origin. These seven parameters uniquely determine the filter as shown in Figure 4. Typically, a large bank of filters is employed for texture analysis tasks. This can be very computationally expensive due the inverse Fourier transform, but as seen through the image sequence textures are stable in terms of frequency over a limited time interval (Macenko, 2007). This time can be used to optimize the Gabor filter bank set to be applied. Hence, reducing the execution time and finding the appropriate filter parameters to detect and estimate the texture motion (Rad et al., 2010). The n filters (n corresponds to the number of moving objects) to be found are encoded as seven bytes each, creating a 7n byte chromosome. Each byte encodes one of the filter parameters (ω_{x0}, ω_{y0}, ω_{t0}, σ_x, σ_y σ_t,θ). A large pool of such chromosomes is then created randomly, each chromosome representing a different subset of filters as shown in Figure 4.

Each filter is fully determined by choosing the seven parameters. Therefore, choosing a filter for a particular application involves optimizing these seven parameters. Assuming that N filters are needed in an application, 7xN parameters need to be optimized. Solving this high dimensional multivariate optimization problem is very difficult in general.

These chromosomes are evolved in a normal manner, using cross-over and mutation operators as described in Yilmaz et al. (2006). Each gene has a particular cost function (fitness) associated to it, which means how well the encoded filters perform when used for a moving texture classification task. This stage is the most computationally expensive part of the entire training process since it creates each Gabor filter and computes its inverse Fourier transform (Kamarainen et al., 2006; Heikkila and Pietikainen, 2006). From the statistics stated earlier, we can compute the genetic fitness function as the following:

$$F_s = \sum_{s=1}^{S} (P_s)^{3/2} \ln\left[\frac{(\omega_x-\omega_{x0})^2+(\omega_y-\omega_{y0})^2+(\omega_t-\omega_{t0})^2}{\sigma_{sx}^2+\sigma_{sy}^2+\sigma_{st}^2}\right] \quad (6)$$

where the low level routine produce S textures in the image and each texture s produced consists of P_s pixels. The aforementioned measure has been shown to correlate well with measures based on ground truth for synthetic images, and has been applied successfully to real images. Thus, the higher the fitness value, the more significant is the identity of the associated chromosome. The GA algorithm is summarized as follows:

(1) Create first random population
(2) Fitness evaluation
(3) While <Termination condition is false> do
(a) Selection (and replacement)
(b) Recombination
(c) Mutation
(d) Fitness evaluation
(4) End while

After shadow removal, we have the binary image of foreground pixels that corresponds to the objects. The next task is to find the separate objects. To accomplish this, we first remove speckle noise by morphology, then determine connected regions, and group regions into separate objects for both foreground pixels and background pixels separately. To speed up the filtering process, a fast erosion-dilation filter is applied. Using the binary foreground/background map, a connected component analysis is subsequently applied to determine the connected regions of the foreground pixels after filtering. This may be implemented as a graph-based single-pass algorithm instead of a conventional two-pass approach. After detecting the connected regions, the total number of pixels of a connected region is easily computed, the center of mass and the object box coordinates.

EXPERIMENTAL RESULTS

We need to find the oriented textures with frequencies (ω_x,ω_y,ω_t) in the spatio-temporal volume in the two 100 x 100 pixels test sequence images shown in Figure 4a and b. The image includes 3 objects which have undergone a 2-D displacement. Then, the plan which holds the loci of the texture corresponding points in the spatio-temporal frequency domain estimates the optical flow (u,v) of each

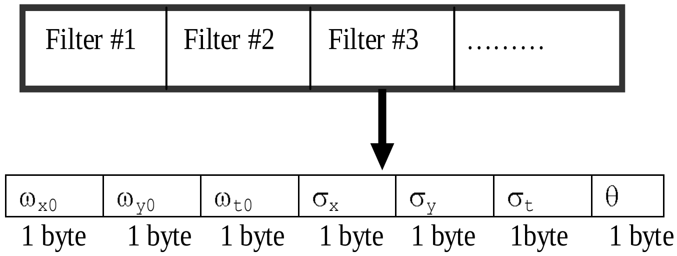

Figure 4. Encoding scheme.

Table 1. Fitness values for different populations.

Pop	(σ_x, σ_y, σ_t)	ω_x	ω_y	ω_t	θ	Energy	Fitness
5	(10,13,3)	9	16	10	12.3	0.9256	1250.2
10	(10,13,3)	9	16	10	13	0.9656	1207.6
20	(10,13,3)	9	15	10	20.	1.1251	1165.8
40	(10,13,3)	9	16	10	19.7	1.1625	1158.0
80	(10,13,3)	10	15	10	15.71	1.0097	1280.1
160	(10,13,3)	10	15	10	14.6	0.9481	1265.9
320	(10,13,3)	10	15	10	16.21	1.0382	1112.7
1000	(10,13,3)	9	15	10	18.52	1.1056	1199.4

Table 2. Some parameter space values for the 3 objects during the detection stage.

Parameter	Image	N°	1	Image	N°	2
	obj 1	obj 2	obj 3	obj 1	obj 2	obj 3
Area (pix)	1603	1501	46	1665	1572	155
Perimeter (pix)	165	127	96	123	132	103
Minimum radius	22	18	3	26	21	5

object. In this case we choose the range of θ values from 12 to 20°. The motion-energy values associated to varying populations are shown in Table 1 and Table 2

In this example the cross-over and mutation probabilities were set to 0.85 and 0.01 and were determined empirically. Initially, the algorithm was run using a large population (1000) and for a few generations (Ramzan et al., 2011). Using the fitness values, the global maximum in the space may be found and concentrated upon by the genetic algorithm. The sudden drop in the fitness value may be attributed to the high variations in other parameters (especially if the values taken are beyond the limits of θ). We notice that at the end of each generation there are a number of chromosomes reflecting spatio-temporal points are neighbors and of similar fitness values. Therefore, the parameter space contains some sets of parameters with nearly similar fitness values and can be selected without a significant loss. The quality of results for population size 40 shows that the optimization can be reached quickly with a very high confidence. Generally, the best parameter values can be found in a training period on an application dependent basis.

In the example in Figure 5, we notice a pure translation in the positive x-direction. After, some low level processing, 3 objects are identified as moving objects in the sequence. The following table gives the parameters linked to each object which has undergone a displacement in the image sequence. The measured 2-D displacement parameters are given by Table 3.

Finally, spatio-temporal filtering process based on GA performs the estimation process of the optical flow as illustrated by Table 4.

Moreover, as λ approaches unity, the object motion has a tendency to be a pure translation (without zoom) in a 2-D plane parallel to the image plane. Accordingly, R(°) reflects the rotation of the objects from one frame to the other. Consequently, precision in the optical flow estimation process has been increased for a several moving object image sequence. This is mainly due to the spatio-temporal filtering optimization using GA. Also, it

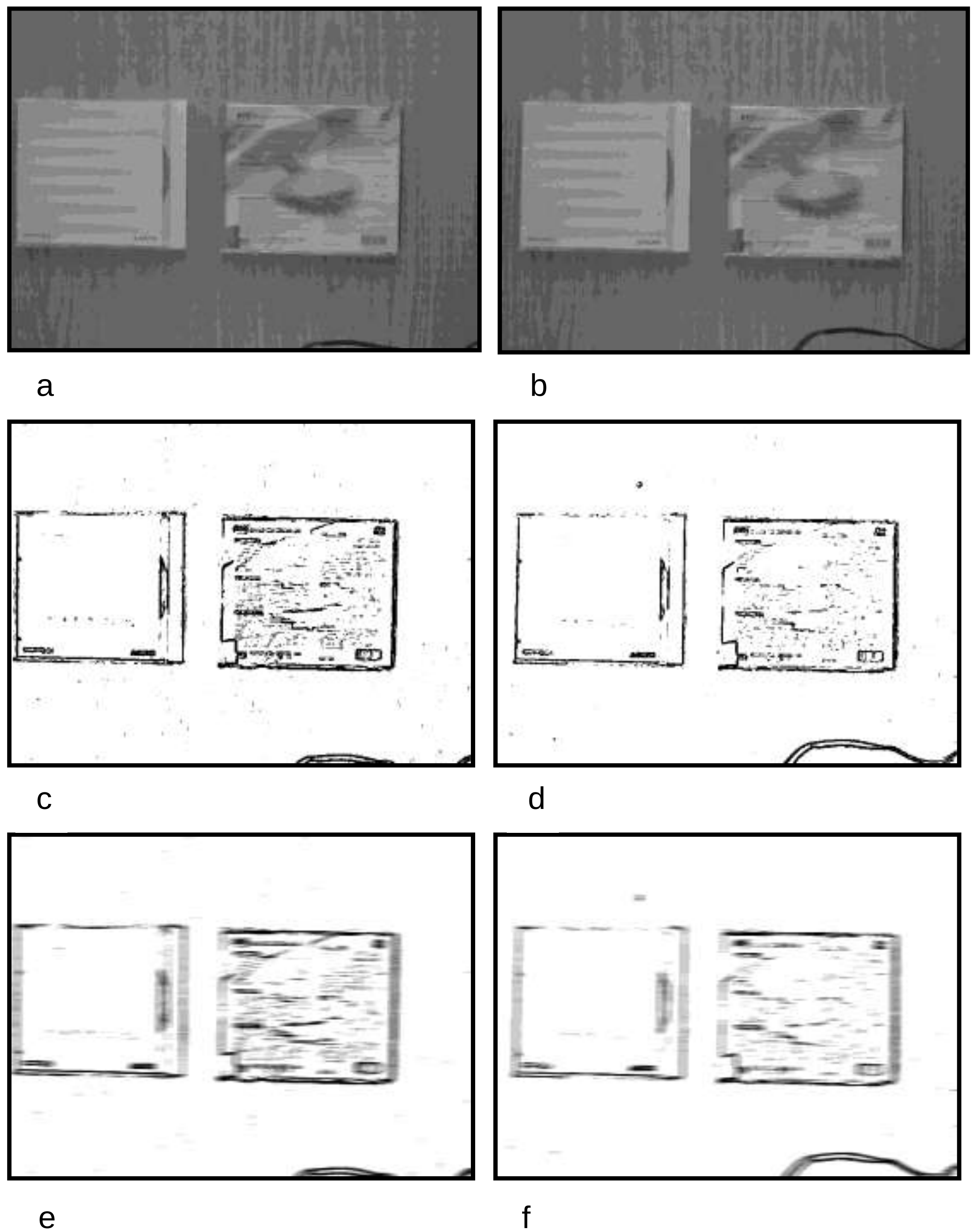

Figure 5. Test images (a, b) images containing 3 objects, (c, d) images after low level processing, (e, f) images after the filtering stage.

Table 3. Apparent displacement parameters for the three objects.

Parameter	Tx (pix)	Ty (pix)	F	R (°)
Object 1	32	7	0,775	2
Object 2	29	8	1,057	1
Object 3	23	3	0,951	2

Table 4. Optical flow estimation for the 3 objects.

Parameter	λ	*u* (pix)	*v* (pix)	R (°)
Object 1	0,951	32	5	0
Object 2	0,964	26	8	5
Object 3	0,958	21	5	8

has reduced the failure rate in motion detection using the classical multi-resolution approaches by at least 20%.

Conclusion

Computationally, GA are becoming completely feasible in a real-time vision system; however, the implicit parallelism they showed initiated very wide research activities. The role of optimization has proved to be very important and efficient in the object motion analysis. Actually, it has an inherent place in vision systems. This is true either in the feedback-based systems where a goal directed approach is applied or in a knowledge-based systems. General spatio-temporal filtering methods used for object motion analysis use the entire filter bank for texture processing, hence increasing the computation time. This paper has described a technique to enhance the spatio-temporal filtering techniques when used in motion analysis. In fact, a general framework for the optimization of the texture classification has been presented in order to estimate the motion parameters of an object through image sequence. The selection of the filter subset is well performed by the GA to provide, in a less computation time, optimized solution motion estimates of the object. Furthermore, spatio-temporal Gabors features provide a compact and convenient method for encoding both texture and motion properties present in time-varying image data. We have shown experimentally, the convergence of the approach that leads to a quicker solution by optimizing with a small Gabor filter set.

REFERENCES

Goldberg A (1989). "Inherent ambiguities in recovering 3-D motion and structure from a noisy field," IEEE Trans. Pattern Anal. Machine Intell., Il: 477-489.

Bebis G, Louis S, Varol Y, Yfantis A (2002). “Genetic object recognition using combinations of views,” IEEE Trans. Evol. Comput., 6: 132-146.

Cornea ND, Silver, D, Min P (2007). Curve-skeleton properties, applications, and algorithms. IEEE Trans. Vis. Comput. Graph.,13(3): 530-554

Dorfmüller-Ulhaas K (2003). Robust optical user motion tracking using a Kalman filter. Technical Report. University of Augsburg.

Elarbi-Boudihir M (2008). Enhanced fuzzy Perception for Robot Vision. J. Digital Info. Manage., 6(6): 490-497.

Heikkila M, Pietikainen M (2006). “A Texture-Based Method for Modeling the Background and Detecting Moving Objects,” IEEE Trans. Patt. Anal. Machine Intell., 28(4): 657-652.

Horn KP, Schunck BG (1981). "Determining optical flow," Artificial Intelligence, 17: 185-203.

Kamarainen JK, Kyrki V, Kalviainen H (2006). Invariance properties of Gabor filter-based features—Overview and applications. IEEE trans. Image Process., 15(5): 1088-1099.

Khelifi S, Elarbi-Boudihir M, Nourine R (2004). Video Database Indexing: An Approach Based on Fuzzy Classification of Moving Objects in Outdoor Videos. Proceedings of 8th Maghrebian Conf. Software Eng. Artificial Intelligence, MCSEAI'04. May 9-12, Sousse Tunisia. ISBN: 9973-37-193-3. 555-566. http://www.tunet.tn/mcseai04/mcseai.pdf

Kim S, Mynuddin M, Park S (2010). Uncertainty analysis of 3D motion data available from motion analysis system. Int. J. Phys. Sci., 5(7): 1191-1199.

Ma L, Chelberg D, Celenk M (2005). Spatio-temporal modeling of facial expressions using Gabor-wavelets and hierarchical hidden Markov models. In IEE International Conference on Image Processing, ICIP Vol. 2.

Macenko M (2007). Lesion detection using Gabor based saliency field mapping. Medical Imaging, Proc. SPIE, 6512, Feb. 17-24, San Diego, CA.

Maunsell J, Ferrera V (1995). Attentional Mechanisms in Visual Cortex. . The Cognitive Neuroscience. M. Gazzina edition. MIT Press, pp 451-461.

Movellan JR (2005). Tutorial on gabor filters. Technical report, MPLab Tutorials, UCSD MPLab.

Petkov N, Subramanian E (2007). Motion detection, noise reduction, texture suppression, and contour enhancement by spatiotemporal gabor filters with surround inhibition. Biol. Cybern., 97(5): 423-439.

Prazdny K (1983). "On the information in optical flows," Computer Vision, Graphics, Image Process., 23: 239-259.

Rad A, Dehghani A, Rehan KM (2010). Vehicle speed detection in video image sequences using CVS method. Int. J. Phys. Sci., 5(17): 2555-2563.

Ramzan M, Qasim KM, Iqbal M, Aasem M, Jaffar A, Anwar S, Adnan A, Tamleek MA, Alam M (2011). A genetic algorithms based approach for conflicts resolution in requirement.Int. J. Phys. Sci., 6(4): 828-836.

Verri A, Poggio T (1998). "Against quantitative optical flow," in Proc. First Int. Conf Computer Vision, Washington, D.C.: IEEE Computer Soc. Press, pp. 171-200.

Wang Y, Papageorgiou M, Messmer A (2006). Renaissance: A real-time freeway network traffic surveillance tool. In IEEE ITSC '06.

Yilmaz A, Javed O, Shah M (2006). Object tracking: A survey. ACM Comput. Surv., 38(4).