Human Faces Detection and Tracking in Video Sequence

Renaud Séguier, Annick Le Glaunec, Bernard Loriferne

École Supérieure d'Électricité, Electronic Department, Rennes, France

Abstract— **This paper presents a system which detects and tracks small faces (20 to 50 pixels wide) in video sequence. A face is modelised by an approximative ellipse, his motion is restricted and linear in a small set of images acquisition. To detect objects whose model is approximately known, we propose a new version of the Hough transform : the Fuzzy Generalised Hough Transform (FGHT). In the transformation step, the votes in parameter space are weighting according to the values of two membership functions. For each pixel, they take into account its position and the angle of its local edge normal. In the detection step in parameter space, the computation for each cell of its level and the mean level of nearness cells allow a robust detection against noise. Performances of this algorithm were evaluated on the basis of several sequences. It appears that if the background is uniform, faces are always well detected. On the other hand, if the background is very complex, the system can loose faces after a certain time.**

Keywords— **Pattern recognition, Face Detection, Face location, Fuzzy Generalised Hough Transform,**

I. Introduction

Face detection in digital pictures is a task which is more and more investigated in recent years. This work overlies many areas of application : automatic face recognition, man-machine interaction systems, T.V. public estimation [1] [2], visual communication systems [3], video-surveillance [4] etc.

This pattern recognition problem has been approached in two ways.

In the first one, a database of face images is compiled to extract the meaningful characteristics of a face.[5] applies a principal component analysis and evaluates the best eigenvectors to create a "face space". To detect a face in an image, [6] calculates the Euclidean distance between the image and its projection into the face space. This distance, compared to a threshold, is used as a measure of the "faceness". This test is not sufficient to detect small faces (less than 50x70 pixels) in a scene. [1], [2] trained an artificial neural network using standard gradient backpropagation algorithm. The database contains face and background images. The neural network is used as a classifier. When an input image is presented, the network must classify it as a face or a background. As a matter of fact, it is very difficult to define what is a background, considered as a "non face". This first approach presents a disadvantage : it is very difficult to choose a relevant database of face images. People wearing moustaches, glasses, hats, bold people, side-faces, color people are detected with difficulty.

In the second approach, a model of a face is defined a priori. [7], [8] localise eye pairs and mouth. A certain geometrical configuration among these features indicates the presence of a face. [9] considers the nose in addition. It is impossible to detect side-face this way. [10] consider the face as a whole unit. The three arcs corresponding to the two sides of the face and the head, are searched and must match with a simple model. To detect the arcs, he applies a Generalised Hough Transform (GHT). The number of faces in the image must be known beforehand. For [11], the outlines of a face can be approached by an ellipse which is searched in an image by performing a GHT. He supposes that it exists only one person in the image. These methods are inadequate for the detection of several small resolution faces.

II. Our approach

We need for the recognition task an algorithm which is able to detect and track few faces in a video sequence. Our system must be sufficiently general to be useful in a large variety of application. The main constraints are :

- the algorithm must be simple enough to be integrated
- the number of faces is unknown
- the heads are in front-view or in profile
- the background is unknown
- the faces' width is small (20 to 50 pixels)

Our method is as follows : the face is considered as a whole unit, the outlines are approached by an ellipse. This one is detected by a Hough transform. We assume that at a given time in a sequence, people is moving. If motion is detected, the system localises the face and track them throughout the sequence.

Of course, the outlines of a face does not describe exactly an ellipse, it is the reason why we perform a Fuzzy Hough Transform (FHT) which can recognise vagueness known patterns. We propose in section 3 a new version of the FHT, the Fuzzy Generalised Hough Transform (FGHT) : the computational complexity is reduced and the detection in parameter space more robust. In section 4 we present the system in detail. After showing some experimental results in section 5, we conclude in section 6.

III. The Fuzzy Generalised Hough Transform (FGHT)

A. The various versions of the Hough transform

The Standard Hough Transform (SHT), proposed by Paul Hough in 1962, is widely used in pattern recognition. To know all about this transformation and its applications, readers can report to the references [12] and [13].

We can note an interesting evolution in the various way of performing a Hough transform.

In SHT, for each edge pixel, a vote is performed in the parameter space. This vote increases uniformly the level

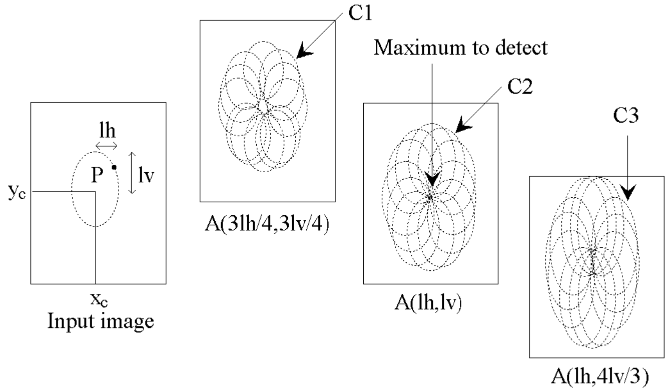

Fig. 1. The Hough transformation of an ellipse

of all cells that can caracterise the pattern's parameters to be recognised, considering the position of the edge pixel.
In the Generalised Hough Transform (GHT), the local edge normal at each edge pixel of the input image is considered. This way, the number of cells taken into account in the parameter space is decreased.
In the Weighted Hough Transform (WHT), the votes in the parameter space are weighted according to the magnitude of the intensity gradient at each pixel.
In the Weighted Generalised Hough Transform (WGHT) [14], the votes are weighted according to the degree of occlusion of each scene feature. By interpreting the weights as membership function values, the WGHT includes notions from fuzzy set theory.
In the Fuzzy Hough Transform (FHT), [15] and [16] proposed to detect objects which are known just through their approximate geometrical model. For each edge pixel, a SHT is performed and the set C of cells to be increased is computed. The cells' votes in the parameter space are weighted according to the Euclidean distance from the set C. The idea is to consider that each edge pixel can belong to a known object Θ, but can also belong to another one whose geometrical properties are not so far from Θ .
Throughout those versions of the Hough transform, we can see the need of labelling the features in the scene to perform the votes in parameter space in a relevant manner and to reduce the computational complexity of the algorithm.

B. The FGHT

Briefly, the principle of the Hough transform is as follows. Considering an input image in which exist an ellipse, we want to find its parameters, say $\{lh, lv, x_c, y_c\}$. For each possible size of the ellipse, we create an accumulator A(lh,lv) which has the same resolution as the input image. The set of accumulators constitute the parameter space.
The point P in the input image can belong to a wide variety of ellipses. In the three accumulators in figure 1 we compute the sets (C1, C2, C3) of possible location of those ellipses, and perform a vote in the designed cells. As we can see in figure 1, when we have considered all the edge pixels, the maximum over all the cells in the parameter space indicates the centre of the ellipse, and its size. To avoid the computation of the location of possible centres in parameters space, we can store those locations, for each edge point, in a matrix M that has the same resolution with the input image. The position and the value of the element $M_{(i,j)}$ indicates the position of the cell whose level is going to be increase by a vote of weight $M_{(i,j)}$.
To detect approximative geometrical objects, we propose a new version of the Hough transform. The main evolutions are :
• the votes in parameter space are weighted considering the position of each edge pixel and the magnitude of its intensity gradient.
• the detection in parameter space considers each cell through its level and the mean level of nearness cells.

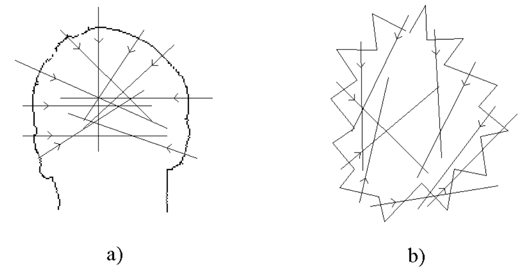

Fig. 2. Different orientations of the normal edge

B.1 The transformation

When the local normal edge is relevant, it is natural to consider this information to reduce the complexity of the Hough transform. For example, in our case, the normal edge point approximately towards the centre of the face (figure 2a), in other case, this information is not interesting (figure 2b).
If we compute a GHT, for each possible value of the normal edge direction evaluated at each edge point we must compute the matrix whose elements are :

$$M_{(i,j)}(lh, lv, \alpha) = \delta(i - x_c, j - y_c) \tag{1}$$

with

$$x_c = x \pm sign(dLx) \frac{lh}{\sqrt{1 + \frac{lv^2}{lh^2} \left(\frac{dLy}{dLx}\right)^2}} \tag{2}$$

$$y_c = y \pm sign(dLx) \frac{lh}{\sqrt{1 + \frac{lh^2}{lv^2} \left(\frac{dLx}{dLy}\right)^2}} \tag{3}$$

x_c and y_c determine the coordinates of the center of the ellipse, lh and lv reprensent the semi-axes. dLx and dLy are the intensity gradient evaluated on each direction on the point P(x,y) (figure 3). α is the angle of the local edge normal and is defined as :

$$\alpha = \arctan\left(\frac{dLy}{dLx}\right) \tag{4}$$

For each point P in the edge image, we consider two cells in the parameter space : c an c' (figure 3) which correspond

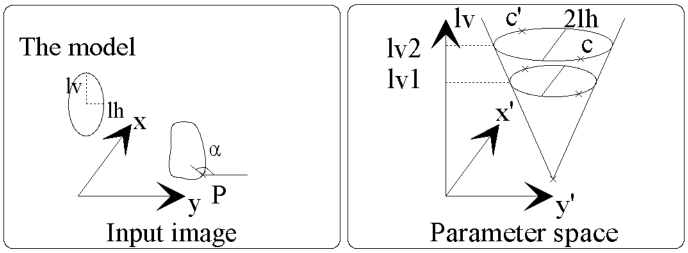

Fig. 3. The two cells considered through the GHT

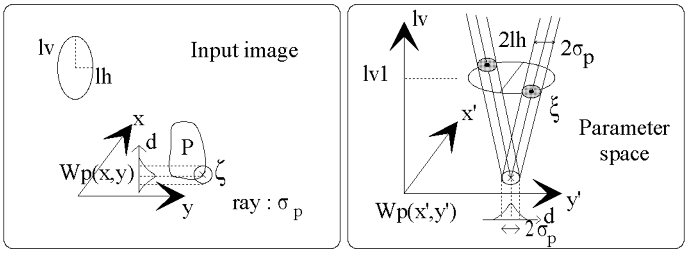

Fig. 4. The two cells' sets considered through the FHT

to the direction indicated by the normal and the inverse one : we do not know if the object to be detected is much dark as the background or inversely.
We now apply the ideas from the FHT : we consider the neighbourhood ζ of the point P (figure 4). For each pixel in ζ, we apply the same GHT, but the votes are weighted by the function Wp(d) which decreases with the distance d between the pixel and the point P.
For each point, we now consider the matrix M' whose elements are :

$$M'_{(i,j)}(lh, lv, \alpha) = \sum_{x'=x-\sigma_p}^{x+\sigma_p} \sum_{y'=y-\sigma_p}^{y+\sigma_p} Wp \cdot M_{(i,j)}(lh, lv, \alpha) \tag{5}$$

with

$$Wp = \begin{cases} K_p \exp\left(-\frac{d^2}{{\sigma_p}^2}\right) & if\ d \leq \sigma_p \\ 0 & if\ not. \end{cases} \tag{6}$$

and

$$d = \sqrt{(x_c - x')^2 + (y_c - y')^2} \tag{7}$$

K_p is a constant chosen empirically, σ_p permits us to incorporate our knowledge about how far the object is from the model.
Applying the same idea when we consider the value of α , we consider the cells which are designated by the GHT with the value α with a high vote and the cells designated by the GHT with nearness values of α with decreasing votes (figure 5).
For each point, we now consider the matrix M'' whose elements are :

$$M''_{(i,j)}(lh, lv, \alpha) = \sum_{\alpha'=-\infty}^{\alpha'=+\infty} W\alpha(\alpha') M'_{(i,j)}(lh, lv, \alpha') \tag{8}$$

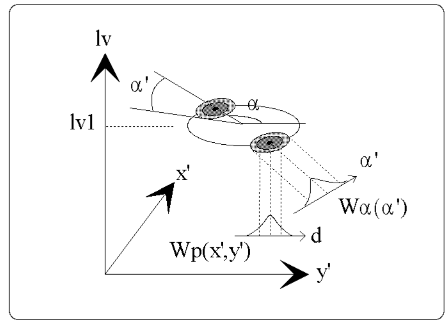

Fig. 5. The effect of the membership functions in the parameter space

with

$$W\alpha(\alpha') = \begin{cases} \exp\left(-\frac{(\alpha-\alpha')^2}{{\sigma_\alpha}^2}\right) & if\ \alpha - \alpha' \leq \sigma_\alpha \\ 0 & if\ not. \end{cases} \tag{9}$$

Like in equation (6), the constant K_α is chosen empirically and σ_α permits us to incorporate our knowledge about how far the normal edge directions of the object are from the normal edge directions of our model.
We have computed the matrix M'' by replacing (x,y) by (0,0) in equations (2) and (3). M'' gives the position of the cells where the vote must be done, relatively to the position of the edge pixel. Now, the value of x_c and y_c which designed the cell to be increased, can be computed after looking the matrix stored in a look up table. Each matrix is attached to a particular size of the ellipse and a specific angle α.
The main steps of the transformation are as follows :
• reset all accumulators
• extract the magnitude of the intensity gradient on each pixel of the input image in the directions of columns (dLx) and rows (dLy)
• for all pixel for which dLx or dLy is upper or equal to a threshold SeuilGrad fixed a priori do the following:
 – evaluate the angle α of the normal edge
 – evaluate the location of the cells for which the level must be increased in the parameter space.
 – perform the accumulator vote:

$$A_{(i,j)}(lh, lv) = A_{(i,j)}(lh, lv) + M''_{(i+x,j+y)}(lh, lv, \alpha) \tag{10}$$

B.2 The detection

In the GHT, the level of the signal in the parameter space is highly relevant. Usually, to find the centre of the ellipses in the image, we keep back the cells whose level oversteps a threshold TLPeak fixed a priori. Each of them constitute a peak Peak(i), which is a candidate to a possible location of the centre of an ellipse. If we know that only one ellipse exists in the image, we simply retain the peak whose level LPeak(i) is the maximum over the N retaining peaks.
When we perform a FGHT, we do not consider a single cell in the accumulators when we perform the transformation, but a set of cells with different weights' vote. For that reason, another parameter can assist the detection : the mean

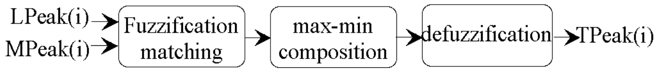

Fig. 6. Architecture of a fuzzy controller

MPeak(i) of the signal in a square region centred on each peak. The main difficulty is to find an aggregation method which can produce a simple test variable TPeak(i) from LPeak(i) and MPeak(i). Compare TPeak(i) to an a priori fixed threshold will permit us to evaluate the membership of Peak(i) to the set "characterise an ellipse in the image". To perform this aggregation, we utilise simple rules from fuzzy expert systems [17] (figure 6).

To yield the max-min composition step simple as possible, the fuzzification matching step must render comparable the two parameters LPeak and MPeak. If a peak Peak(i) caracterises an ellipse in the image, its level LPeak(i) must be among the higher over the N retaining peaks' level. Similarly, MPeak(i) must be among the higher over the N MPeak values. The fuzzification matching step consists in evaluating, for each peak Peak(i), two values $\mu_{LPeak}(i)$ and $\mu_{TPeak}(i)$ which denote the degree to which Peak(i) is a member of the set "characterise an ellipse in the image". Their expressions are :

$$\mu_{LPeak}(i) = \frac{LPeak(i)}{\max\{LPeak(j)_{j=1\ldots N}\}} \tag{11}$$

$$\mu_{MPeak}(i) = \frac{MPeak(i)}{\max\{MPeak(j)_{j=1\ldots N}\}} \tag{12}$$

These two parameters are comparable and normalised, we can execute a simple max-min composition which expresses the variable TPeak as in equation (13).

$$TPeak(i) = \mu_{LPeak}(i) + \mu_{MPeak}(i) \tag{13}$$

The defuzzification step is simple. Considering that the peak which characterise with the higher possibility an ellipse must produce the higher value of TPeak, we simply compare all the values of TPeak(i) over the N Peaks to a threshold to find the centres of the ellipses in the input image.

The main steps of the detection are as follows:
- evaluate the signal in the accumulators through a square window W of width Lc
- if the level of the middle window cell c is greater than TLPeak create the parameter Peak(i) with attached values LPeak(i) and MPeak(i)
  - LPeak(i) is the level of c
  - MPeak(i) is the sum of all cells' level in W
- store the maximum values of LPeak and MPeak
- when all cells are observed, compute and store the N values TPeak(i)
- retain the peaks for which the value TPeak are greater than the threshold TTPeak

IV. The system

In most applications, the room which is filmed by the video camera is empty. After a time, people enter the room and stay in the field of the video. Our system is organised in four modules as we can see in figure 7.

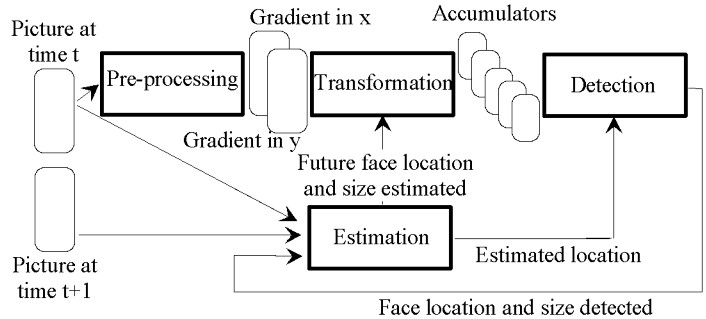

Fig. 7. Architecture of the system

A. Pre-processing

The pre-processing unit extracts the value of the intensity gradient (dLx,dLy) on each pixel of the input image by performing a Shen-Castan filter [18], which is based on Canny's principles [19].

B. Estimation

This unit has several things to do:
- Evaluation of the motion in the left part of the scene. If a person enters the room, the unit detects the motion by a simple difference between the left part of the input image at time $(t-1)$ and (t).
- Estimation of the position (x, y) and the width (wf) of the faces. If a person has been located three times ago with the positions and size $(x_{t-3}, y_{t-3}, wf_{t-3})$, $(x_{t-2}, y_{t-2}, wf_{t-2})$, $(x_{t-1}, y_{t-1}, wf_{t-1})$ the module estimates the future characteristics of the face $\left(\widehat{x}_t, \widehat{y}_t, \widehat{wf}_t\right)$ as:

$$\widehat{x}_t = \frac{3 \cdot x_{t-1} - x_{t-3}}{2} \tag{14}$$

$$\widehat{y}_t = \frac{3 \cdot y_{t-1} - y_{t-3}}{2} \tag{15}$$

$$\begin{array}{c} if\ wf_{t-1} = wf_{t-2}\ or\ wf_{t-1} = wf_{t-3} \\ then\ \widehat{wf}_t = wf_{t-1} \\ elseif\ wf_{t-2} = wf_{t-3} \\ then\ \widehat{wf}_t = wf_{t-2} \\ else\ \ \widehat{wf}_t\ is\ unknown \end{array} \tag{16}$$

- Estimation of the region where must exist a face.
  - if a motion has been detected in the left part of the image, this particular region must be analysed by the next units which must estimate if a face entered the room or not.
  - if a face has been detected for at least three times, the region around the estimated location of the face $(\widehat{x}_t, \widehat{y}_t)$ in a rectangular window of size lxL is considered by the next units with:

$$l = 2 \cdot wf_{t-1}\ and\ L = 3 \cdot wf_{t-1} \tag{17}$$

  - if a face has been detected no more than two times, the region around the last detected position of the face is

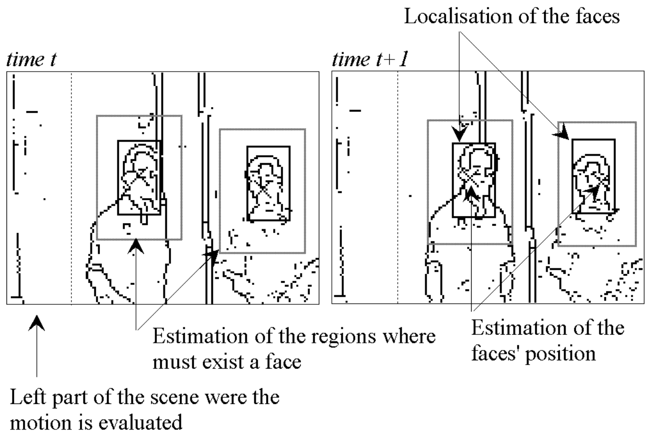

Fig. 8. Ereas evaluated by the estimation unit

considered in a rectangular window of size lxL with:

$$l = 2 \cdot wf_{max} \ and \ L = 3 \cdot wf_{max} \tag{18}$$

considering that wf_{max} is the maximal width of the faces to be detected.

By considering the motion, we solve the difficult problem of multiple faces detection in an image: the task is now restricted to the detection of one face in several regions of the image.

C. Transformation

To reduce the number of accumulators, we link the parameters lh and lv altogether. Usually the ratio $R = \frac{lv}{lh}$ for the outlines of a face modelled by an ellipse, went from $R_{min} = 1.2$ to $R_{max} = 1.7$. For that reason, for each value of lh, we create an accumulator A(lh) which is the summation of all accumulators attached to a specific value of (lh,lv) for which:

$$lv \in \{lh \cdot R_{min}, lh \cdot R_{min} + 1, ..., lh \cdot R_{max} - 1, lh \cdot R_{max}\} \tag{19}$$

The faces' size to be detected is restricted to the interval $[wf_{min}, wf_{max}]$ with $wf_{min} = 2 \cdot lh_{min}$ and $wf_{max} = 2 \cdot lh_{max}$. To permit a limited number of accumulators, we discretise the values of lh with a step lh_{step}.

$$lh \in \{lh_{min}, lh_{min} + lh_{step}, ..., lh_{max} - lh_{step}, lh_{max}\} \tag{20}$$

Each Accumulator A(wf) is now the summation of multiples accumulators for whose:

$$lh \in \left\{\frac{wf}{2} - lh\varepsilon, ..., \frac{wf}{2} + lh\varepsilon\right\} \tag{21}$$

This can be done through the evaluation of matrix M''':

$$M'''_{(i,j)}(wf, \alpha) = \sum_{lh=\frac{wf}{2}-lh\varepsilon}^{\frac{wf}{2}+lh\varepsilon} \sum_{lv=lh \cdot R_{min}}^{lh \cdot R_{max}} M''_{(i,j)}(lh, lv, \alpha) \tag{22}$$

For fixed values of lh_{step} and $lh\varepsilon$, different accumulators take into account different numbers of ellipses' size. To render the peaks level comparable between accumulators, we normalise the weights produced in matrix M''' by the number of pairs (lh,lv) which are taken into account in the matrix.

We do not need a precise evaluation of the normal edge so we can strongly quantify the angle α with a step α_{step}, to reduce the number of accumulators. α can take the following values:

$$\alpha \in \left\{0, \alpha_{step}, ..., \frac{\pi}{2}\right\} \tag{23}$$

Because of this strong quantification, we must change the form of Wα from equation (9). We propose now:

$$W\alpha(\alpha') = \begin{cases} K_\alpha \exp\left(-\frac{(\alpha' - (\alpha - L_\alpha))^2}{\sigma_\alpha{}^2}\right) & if \ \alpha' \leq \alpha - L_\alpha \\ K_\alpha \exp\left(-\frac{(\alpha' - (\alpha + L_\alpha))^2}{\sigma_\alpha{}^2}\right) & if \ \alpha' \geq \alpha + L_\alpha \\ K_\alpha & if \ not. \end{cases} \tag{24}$$

In the transformation unit we perform the following actions:

- reset all accumulators
- for all pixel for which dLx or dLy is greater or equal to SeuilGrad:
  - evaluate the angle $\alpha = \arctan(dLy/dLx)$ of the normal edge
  - evaluate the cells location and the weight of the votes by considering the matrix attached to the angle previously calculated
  - perform the vote in each accumulator

D. Detection

From the estimation unit, we know the estimated position of the face $(\widehat{x}_t, \widehat{y}_t)$. For each peak located in $(X_{Peak(i)}, Y_{Peak(i)})$ we compute an estimated error δ

$$\delta\,(Peak(i)) = \sqrt{\left(\widehat{x}_t - X_{Peak(i)}\right)^2 + \left(\widehat{y}_t - Y_{Peak(i)}\right)} \tag{25}$$

We perform the steps presented in III-B.2, except that we only take into account the peaks for which

$$\delta\,(Peak(i)) \leq wf_{t-1} \tag{26}$$

Because we must find only one ellipse in an observed region, we retain the peak which gets the maximum value of TPeak.

V. The results

We have tested our system on multiple sequences (sample rate : 13 images per seconds). The size of an image in the sequence is 160x120 pixels which is sufficient to detect and track few faces in the same room.

We have chosen the following values:

Fig. 9. Detection of a side-face

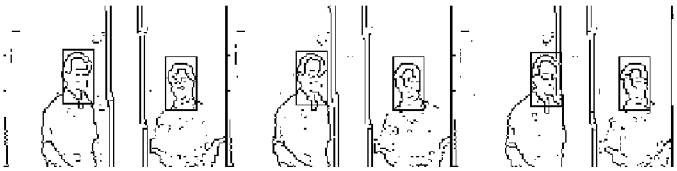

Fig. 10. Two faces are detected : the first one is motionless, the second one is in movement

$SeuilGrad = 16$	$R_{min} = 1.2$	$R_{max} = 1.7$	$Lc = 5$
$lh_{min} = 11$	$lh_{max} = 23$	$lh_{step} = 3$	$lh\varepsilon = 1$
$\sigma_\alpha = \frac{\pi}{50}$	$L_\alpha = \frac{11 \cdot \pi}{70}$	$K_\alpha = 5$	
$\sigma_p = 4$	$K_p = 5$	$TLPeak = 60$	

We have used integer values for Wp and Wα. The matrix values are stored on four bits.
We get now five accumulators which permit us to detect faces whose width ranged from twenty to fifty pixels.

A. The performances of the system

Because the algorithm is based only on the outlines of the face, the system is able to detect side-face, as we can see in figure 9.
The condition in equation (26) is not too strict: we can detect and track faces in movement or motionless as we can see in figure 10.
We have added white noise to the intensity gradient. We can see in figure 11 how the system is robust against that particular type of noise.
If the background is uniform, the algorithm is sufficiently robust to detect and track persons wearing hat (figure 12).
The matrix normalisation exposed in IV-C works well since the system is able to track a face when it grows up : four

Fig. 11. The robustness of the FGHT against noise

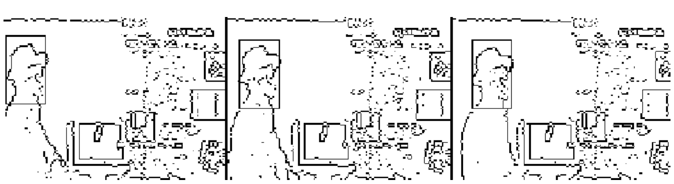

Fig. 12. Detection of a person wearing a hat

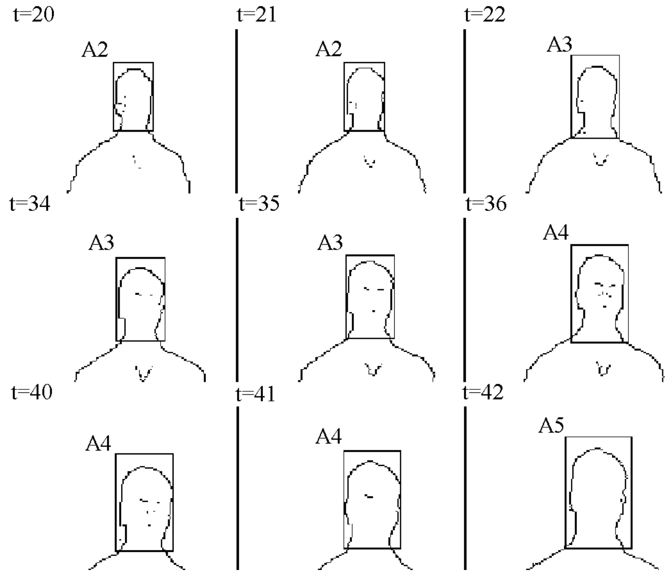

Fig. 13. Different accumulators selected during the same sequence

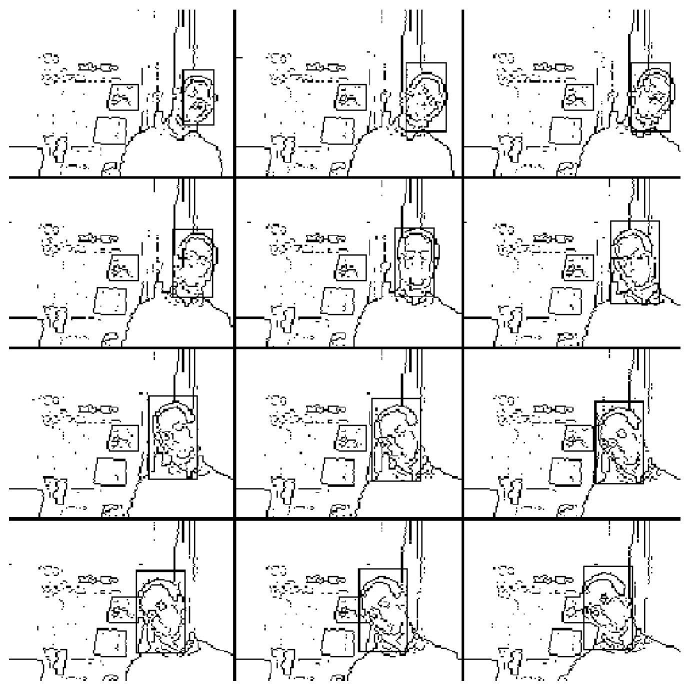

Fig. 14. Detection of leaning faces

Fig. 15. Detection in a complex background

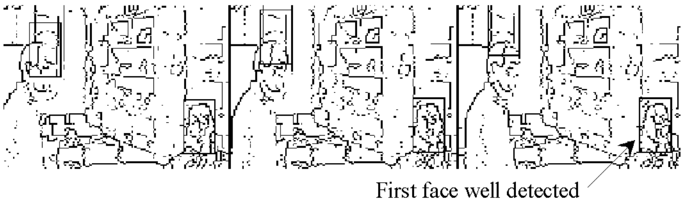

Fig. 16. Wrong detection in a complex background

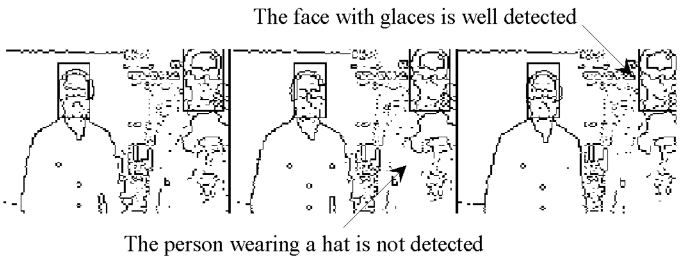

Fig. 17. Loss of the person wearing a hat

of the five accumulators are selected when the face came to the lens (figure 13).
The fuzzy transformation permits us to detect faces even if they are not vertical, as we can see in figure 14 .
When the background is complex, the system can detect faces if they are well contrasted (figure 15).

B. The system's limits

Of course, our system has several limitations.
Considering the sequence of figure 15 where the background is complex, if a face is not sufficiently contrasted, the localisation failed (figure 16).
It is also impossible to detect a person wearing a hat if the background is not uniform (figure 17).
The system can localise and track few faces, even if they are close at hand, but if they intercross themselves, the system focuses its attention on the face which get the most contrasted outline (figure 18).

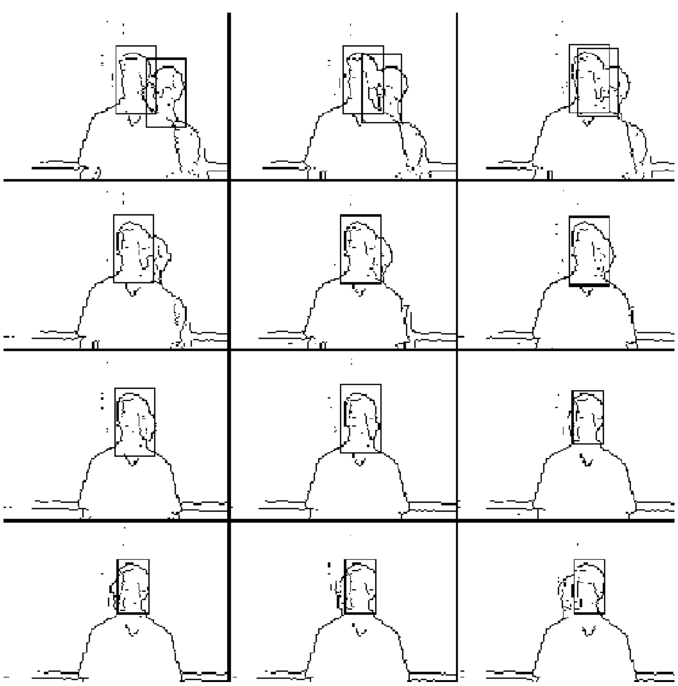

Fig. 18. Detection of close at hand faces

VI. Conclusion

We have presented a new version of the Hough transform : the Fuzzy Generalised Hough Transform (FGHT) which is able to detect object modelised by an approximate curve. The FGHT was integrated in a system to detect and track small faces in video sequence.
If the background is uniform, faces are always well detected. On the other hand, if the background is very complex, the outlines of the faces must be well contrasted to permit their localisation in the image.
This algorithm is simple enough to be integrated, and then allows detection and tracking of small faces, even side ones, in real time (sequence's sample rate : 25 images per seconds). An electronic card which integrates this algorithm is in progress. Programmable components Altera (Flex8000) and DSP Texas Instruments (TMS320C50) are exploited.

References

[1] G.Burel D.Carel, “Détection et localisation de visages sur image numerique”, *Revue technique Thomson-CSF*, 1993.

[2] R.Vaillant C.Monrocq Y.LeCun, “Localisation de visages dans des images”, *Revue technique Thomson-CSF*, 1993.

[3] T.I.P.Trew R.D.Gallery D.Thanassas, “Automatic face location to enhance videophone picture quality”, in *British Machine Vision Conference*, 1992.

[4] J.L.Perry J.M.Carney, “Human face recognition using a multilayer perceptron”, in *Internatinal Joint Conference on Neural Network*, 1990.

[5] M.Kirby L.Sirovich, “Application of the karhunen-loéve procedure for the characterization of human faces”, *Transactions on PAMI IEEE*, vol. 12, 1990.

[6] M.Turk A.Pentland, “Eigenfaces for recognition”, *Journal of cognitive Neuroscience*, 1991.

[7] P.Seitz M.Bichsel, “”the digital doorkeeper” -automatic face recognition with the computer”, in *IEEE Intern. Carnahan Conf. on Security Technology*, 1991.

[8] G.Chow X.Li, “Towards a system for automatic facial feature detection”, *Pattern Recognition*, vol. 26, no. 12, 1993.

[9] G.Yang T.S.Huang, “Human face detection in a complex background”, *Pattern recognition*, vol. 27, no. 1, 1994.

[10] V.Govindaraju S.N.Srihari D.Sher, “A computatinal model for face location based on cognitive principles”, in *Tenth national conference on artificial intelligence*, 1992.

[11] J.Korosec L.Gyergyek N.Pavesic M.Vezjak A.Gere, “Face contour detection based on the hough transformation”, *Automatika*, vol. 31, 1990.

[12] J.Illingworth J.Kittler, “A survey of the hough transform”, *Computer vision, graphics, and image processing*, 1988.

[13] V.F.Leavers, “Survey, which hough transform ?”, *Computer vision, graphics, and image processing*, vol. 58, no. 2, 1993.

[14] S.M.Bhandarkar, “A fuzzy probabilistic model for the generalized hough transform”, *IEEE Trans on systems, man, and cybernetics*, 1994.

[15] K.P.Philip E.L.Dove D.D.McPherson N.L.Gotteiner W.Stanford K.B.Chandran, “The fuzzy hough transform-feature extraction in medical images”, *IEEE Transactions on medical imaging*, vol. 13, no. 2, 1994.

[16] J.H.Han L.T.Koczy T.Poston, “Fuzzy hough transform”, *Pattern Recognition Letters*, , no. 15, 1994.

[17] W.Pedrycz, *Fuzzy Control and Fuzzy Systems*, Research Studies Press LTD., 1993.

[18] J.Shen S.Castan, "An optimal linear operator for step edge detection", *Graphical Models and Image Processing CVGIP*, vol. 54, no. 2, 1991.

[19] R.Deriche, "Using canny's criteria to derive a recursively implemented optimal edge detector", *International Journal of Computer Vision*, 1987.