Prediction mechanism for the face tracking algorithm

Abstract. The paper presents a modified version of the algorithm dedicated for face tracking in video stream. The detection procedure involves gradient technique and the prediction of the face coordinates by the modified adaptive Kalman filter. The proposed combination of face contour detection and coordinates extraction can be used in modern interfaces to facilitate user interaction with the computer.

Introduction

With the accelerating growth of information technology, growing interest in new techniques of human interaction with computers and machines (HCI). One such example is the idea of a traffic control in a virtual 3D scene by tracking the position of the observer's face. Known in the literature, face-tracking algorithms utilize two basic modules: gradient detection module and matrix of hits also referred to as the hit-matrix which is the certain form of the Hough's accumulator space. In this paper we will try to improve the speed up hit-matrix computations by using Kalman filter.

Assumed edge detection method

The first stage of a classical algorithm for face detection is to determine the matrix of gradients, which will include local changes of illumination in the image obtained in early processing step. Besides determination of the gradients matrix in the vertical and horizontal directions, the suitable gradient matrices for angles of 45o and 135o is calculated, as shown on the Fig.1.

![Fig. 1: Calculated gradients (a) only for the vertical axis (b) with respect to the axis of skew.](image-url)

Gradient matrices are determined as follows:

1. \( G_{x,y} = P_{y,x} - P_{y,x+1}, \)
2. \( G_{45,y,x} = P_{y-1,x} - P_{y+1,x+1}, \)
3. \( G_{135,y,x} = P_{y-1,x+1} - P_{y+1,x-1}, \)

where: \( P_{y,x} \) - the brightness value at the position \([y,x]\),

\( G_{x,y} \) - horizontal gradient value at the considered point \( P_{y,x} \),

\( G_{45,y,x} \) - value of the vertical gradient at the point \( P_{y,x} \),

\( G_{135,y,x} \) - value of the gradient at an angle 45° in the point \( P_{y,x} \),

\( G_{135,y,x} \) - value of the gradient at an angle 135° in the point \( P_{y,x} \). For each point of the image we can calculate the length of the vector gradient, using the following formula:

\[
G_{x,y} = \sqrt{(G_{x,y})^2 + (G_{y,y})^2},
\]

and similarly for pairs of inclined gradients:

\[
G_{2,y,x} = \sqrt{(G_{45,y,x})^2 + (G_{135,y,x})^2}.
\]

The points, where the length of the vector \( G_x \) or \( G_y \) is less than some empirically accepted values can be excluded from further calculations. We can also assume that these points are caused by noise and do not belong to the edges in the image. This is similar circumvention known as the adaptive manner of edge detection used by Canny-Deriche edge detector. The idea behind the procedure of searching for the face plant of the selected point presents Fig.2.

![Fig. 2: Determination of the circle center containing the face for the matrix of perpendicular gradients \( G_x \) and \( G_y \) respectively.](image-url)

Description of the standard terminology that is used to describe consecutive symbols: \( P \) - currently analyzed image point, \( Sy,x \) - the probable measure of oval face center we are searching for, \( R \) - expected radius of the contours of observed face (in the figure marked as \( R_o \)), \( Gy,x \) - gradient vector at point \( P \) (which is determined from the values of gradients in the matrices \( G_{x,y} \) and \( G_{y,y} \)), \( Gy\'y,x \) - horizontal and vertical component of the \( R \) vector. Subsequently the formulas for the coordinates of the circle \( S \) can be derived:

\[
R_o^2 = (Gx)^2 + (Gy)^2, \quad G = (Gx)^2 + (Gy)^2,
\]

where:

\[
G = G \cdot \frac{Gy'}{R_o}, \quad G = G \cdot \frac{Gy}{R_o}.
\]
After the successive transformations we obtain[2]:

\[
G_x' = \frac{R_o G_x}{\sqrt{G_x^2 + G_y'^2}}, \quad G_y' = \frac{R_o G_y}{\sqrt{G_x^2 + G_y'^2}}.
\]

Eventually, searched coordinates of the circle S are formulated as:

\[
S = (x + G_x', y + G_y'),
\]

where: \(x, y\) - coordinates of the analyzed point.

A similar analysis for the \(G45\) and \(G135\) gradients Fig. 3 shows.

**Fig. 3. Determination of the circle containing the face for the gradient matrix pair \(G45\) and \(G135\)**

Assumed description: \(P\) - currently analyzed image point, \(S2_{y,x}\) - the predicted measure for an oval face we are searching for, \(2_oR\) - expected radius of the oval face, \(G_{2y,x}\) - gradient vector at the point \(P\) (which can be determined from the values of gradients in the matrices \(G45\) and \(G135\) respectively), \(G45'\) - component of the inclined axis 45° for \(R\) vector, \(G135'\) - component of the inclined axis 135° for \(R\) vector too. On the basis of of Fig.3 we can derive formulas for the coordinates of the circle center \(S2\):

\[
R_o^2 = (G45')^2 + (G135')^2,
\]

\[
G45 = \frac{G45'}{G_2}, \quad G135 = \frac{G135'}{G_2},
\]

Eventually, after transformations we obtain:

\[
G45' = \frac{R_o \cdot G45}{\sqrt{G45^2 + G135^2}},
\]

\[
G135' = \frac{R_o \cdot G135}{\sqrt{G45^2 + G135^2}}.
\]

Sought coordinates of the circle center \(S2\) are:

\[
S2 = \left( x + \frac{G45'}{\sqrt{2}}, \frac{G135'}{\sqrt{2}}, y + \frac{G45'}{\sqrt{2}} - \frac{G135'}{\sqrt{2}} \right),
\]

In the language of image processing, these operations can be interpreted as a convolution of image with the modified Prewitt kernels:

\[
f_{Gx} = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & -1 \\ 0 & 0 & 0 \end{bmatrix}, \quad f_{Gy} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & -1 & 0 \end{bmatrix}.
\]

In the initial experiments were used as input images free images from the database of images VIS_DB [3]. Thus obtained gradient map stands the input information to the determination of hit-matrix stage in which the work parameter algorithm will be adopted value of the radius \(R\).

**Stabilized hits matrix**

The classic version of the parametric Hough transform uses an array of so-called accumulators of best matches. In this paper, we use a similar mechanism called the hit-matrix. For the localized coordinates of circle center \(S\) and \(S2\) the increment of matrix elements values at the coordinates of the designated zones, whose size is equal to the size of the analyzed image is made. The initial values of this matrix are zero. The way in which elements values of the hit-matrix are increased and its scope presents Fig.4. The matrix is modified by the values indicated in this figure, at positions of the found point \(S\) or \(S2\) and its eight neighbors. Since the calculated coordinates in a video stream may be subject to some small error due to optical or Salt & Pepper noise and momentary changes in scene lighting conditions, we assumed weighted mechanism for updating the values of the hit-matrix. Incrementing in a fuzzy manner values of the matrix elements provides a stabilizing effect of a sequence of computational results. Example array of hits, which was constructed for a certain portion of the video stream is illustrated in Fig.5.

![Fig. 4. Neighborhood function for hit-matrix updating.](image-url)

![Fig. 5. Sample hit matrix in the 3D view.](image-url)

To increase the speed of the algorithm it was assumed that the hit-matrix will be constructed for the selected frames only, while between them, the face tracking prediction will be conducted.
Kalman filter in the task of motion estimation

Kalman filter is a recursive data processing algorithm, which makes state estimation of dynamical system distorted by noise [4]. We define system state vector \( x \) consisting of \( n \) variables that describe essential properties of the system. Kalman filter addresses the general problem of state estimation process \( x \in \mathbb{R}^n \), which is controlled by a stochastic differential equation:

\[
(17) \quad x_k = Ax_{k-1} + Bu_{k-1} + w_{k-1},
\]

with the use of measurement \( z \in \mathbb{R}^m \)

\[
(18) \quad z_k = Hx_k + v_k.
\]

Random variables \( w_k \) and \( v_k \) respectively denotes the noise of process and measurement. It is assumed that they are independent from each other and correspond to the noise of process and measurement. It is assumed that they are completely compatible. The matrix \( A \) of dimensions \( n \times n \) involves an optional excitation signal of state (control) \( u \)

\[
(20) \quad p(x_k | x_{k-1}) = N(Ax_{k-1} + Bu_{k-1}, Q), p(z_k | x_k) = N(Hx_k, R).
\]

Conditional probability for the above model can be written as follows [4]:

\[
(21) \quad e_k^- = x_k - \hat{x}_k^- , e_k = x_k - \hat{x}_k .
\]

and covariance estimation error a priori and a posteriori as:

\[
(22) \quad P_k^- = E[e_k^- e_k^-]^T , P_k = E[e_k e_k]^T
\]

Finally, we can rewrite a certainty a priori and a posteriori of the estimated state as:

\[
(23) \quad Bel(x_k^-) = N(\hat{x}_k^-, P_k^-), Bel(x_k) = N(\hat{x}_k, P_k).
\]

And the conditional probability for the state \( x_k \) based on the measurement \( z_k \):

\[
(24) \quad P(x_k | z_k) = N(\hat{x}_k, P_k).
\]

In this paper, to take advantage of the Kalman filter, we began by finding the equation that calculates the estimation of a posteriori state \( \hat{x}_k \) as a linear combination of estimation a priori state \( \hat{x}_k^- \) and the weighted difference between the measurement \( z_k \) and prediction of measurement \( H\hat{x}_k^- \) shown below:

\[
(25) \quad \hat{x}_k = \hat{x}_k^- + K(z_k - H\hat{x}_k^-).
\]

The \( z_k - H\hat{x}_k^- \) difference is also referred to as innovation or residual. This residual reflects the discrepancy between prediction of measurement \( H\hat{x}_k^- \) and the actual measurement \( z_k \). More over, residual with a value of zero means that they are completely compatible. The matrix \( K \) of dimensions \( n \times m \) is the coefficient of amplification, which minimizes the covariance error a posteriori. One of the possible solution for \( K \) we defined as:

\[
(26) \quad K_k = P_k^- H^T (HP_k^- H^T + R)^{-1} = \frac{P_k^- H^T}{HP_k^- H^T + R}
\]

The equations of the time update phase can be written as:

\[
(27) \quad \hat{x}_k^- = A\hat{x}_{k-1}^- + Bu_{k-1}, P_k^- = AP_{k-1}A^T + Q,
\]

where \( \hat{x}_k^- \) and \( P_k^- \) means the estimated value of the state and covariance a priori, \( \hat{x}_{k-1}^- \) and \( P_{k-1} \) the optimum estimated a posteriori values obtained in the previous step.

Practical implementation of the Kalman filter applied in presented application

Designing of the model system for the filtration of facial movement can be summarized as follows: model of the system and matrix of state becomes:

\[
(28) \quad S_k = S_{k-1} + w_{k-1}, x = \begin{bmatrix} S_x \\ S_y \end{bmatrix}
\]

Hence the system, the measurement, uncertainty of the system and measurement matrices will be of the form:

\[
(29) \quad A = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, H = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, Q = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, R = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}
\]

To ensure the good efficiency of signal tracking procedure we extended the mathematical model of the system to comply the motion velocity:

\[
(30) \quad \dot{x}_k = \dot{x}_{k-1} + w_{k-1} dt, \dot{V}_k = \dot{V}_{k-1} + V_{k-1} dt
\]

The state matrix we can rewrite in the form:

\[
(31) \quad x = [S_x, S_y, F, V_x, V_y]^T
\]

Regarding above we assumed the final form of the state matrix \( A \)

\[
(32) \quad A = \begin{bmatrix} 1 & dt & 0 \\ 0 & 1 & dt \\ 0 & 0 & 1 \end{bmatrix}
\]
where: \(dt\) - denotes time slice update between consecutive cycles. Measurement matrix, because of the measurable values \(S_x\) and \(S_y\) is written as:

\[
H = \begin{bmatrix}
1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0
\end{bmatrix}
\]

Matrices of systems uncertainty and measurement will be of the following form:

\[
Q = \begin{bmatrix}
q_S & 0 & 0 & 0 \\
0 & q_S & 0 & 0 \\
0 & 0 & q_F & 0 \\
0 & 0 & 0 & q_V
\end{bmatrix}, \quad R = \begin{bmatrix}
1 & 0 \\
0 & 1
\end{bmatrix}
\]

The module for the radius value control we can simple rewrite as:

\[
F_k = F_{k-1} + w_{k-1} \cdot x = \begin{bmatrix} F \end{bmatrix}, \quad A = \begin{bmatrix} 1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1
\end{bmatrix}, \quad H = \begin{bmatrix} 1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 \\
0 & 0 & 1 & 0
\end{bmatrix}, \quad R = \begin{bmatrix} r_S & 0 & 0 \\
0 & r_S & 0 \\
0 & 0 & r_F
\end{bmatrix}
\]

In the final stage, Kalman filter used in the application take the form as follow and the state matrix becomes:

\[
x = \begin{bmatrix}
S_x \\
S_y \\
F \\
V_x \\
V_y
\end{bmatrix}
\]

Thus eventually we can rewrite following formulas:

\[
A = \begin{bmatrix}
1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1
\end{bmatrix}, \quad H = \begin{bmatrix} 1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 \\
0 & 0 & 1 & 0
\end{bmatrix}, \quad Q = \begin{bmatrix}
q_S & 0 & 0 & 0 \\
0 & q_S & 0 & 0 \\
0 & 0 & q_F & 0 \\
0 & 0 & 0 & q_V
\end{bmatrix}, \quad R = \begin{bmatrix} r_S & 0 & 0 \\
0 & r_S & 0 \\
0 & 0 & r_F
\end{bmatrix}
\]

We chosen empirically the following values of coefficients, also assumed as the default EyeTracker application settings: \(q_S = 0.1, q_F = 0.5, q_V = 1, r_S = 5, r_F = 5\). The effect of the Kalman filter simulation for these parameters can be seen in Fig. 6. However, the EyeTracker offers the ability to edit the matrix \(Q\) and \(R\) in the course of its operation, so these parameters can be changed. Fig. 7 shows examples of recorded video sequences and extracted frames with marked both real and estimated face coordinates.

**Conclusions**

Based on the performed tests we can formulate following conclusions. It is possible to create system enabling location of the facial image in real time and use this information to control the virtual world. The proposed model used in this study fulfills its role as a fast algorithm and also the exact location of the face tool in the video stream. Important advantage of the filter is its low computational complexity compared to alternative solutions (such as particle filters). Natural development of the conducted experiments could be elaboration of a mechanism for adaptation this method in Field Programmable Gate Array environment.

Fig. 7. Sample simulation results

**REFERENCES**


**Authors:** dr Zbigniew Gomolka, dr inż. Bogusław Twarog, mgr Paweł Kruty, Uniwersytet Rzeszowski, Instytut Techniki, Zakład Elektrotechniki i Informatyki, ul. Rejtana 16c, 35-959 Rzeszów, E-mail: zgomolka, btwarog, pkrut@univ.rzeszow.pl