Kalman filter for skin-colored object tracking

H.-AGUIRRE-RAMOS, J. G.†, AVINA-CERVANTES, E. and ILUNGA, Mbuyamba

Universidad de Guanajuato, Division de Ingenierias Campus Irapuato-Salamanca., Carretera Salamanca-Valle de Santiago km 3.5 + 1.8 km. Comunidad de Palo Blanco, C.P. 36885. Salamanca, Gto., Mexico.,

Received July 20, 2015; Accepted October 21, 2015

Abstract

Applying a Kalman filter, a skin-colored object is trac- king even if it is partial or total blocked. Using the EM algorithm to obtain the probability distributions for both skin and nonskin classes a naive Bayesian classifier is used to extract the current object position features, and use them as an input for the Kalman filter. Using the skin segmentation instead of the gradient differences, only skin-colored objects are tracked, preventing the tracking of undesired moving objects.

Kalman, expectation maximization, bayes, object tracking.

Citation: H.-AGUIRRE-RAMOS, J. G., AVINA-CERVANTES, E. and ILUNGA, Mbuyamba. Kalman filter for skin-colored object tracking. ECORFAN Journal-Democratic Republic of Congo 2015, 1-1: 10-15

† Researcher contributing first author.

Introduction

Object tracking is an important issue in tasks such as surveillance, biological, control or military processes and other industrial applications such as quality control or mass production. Since one scene can contain several moving objects a simple difference of gradient images can not be used if only one kind of object is required to be tracked; therefore, a more optimal solution to extract the position features is needed.

We have several feature extraction techniques available, but in this paper, the use of the naive bayesian clasifier is explored. The addition of a coste function and the combination of different features show a better performance once they are applied on the database.

Using the UC Irvine Machine Learning Repository skinnonskin database [1], a set of 183792 samples of skin and nonskin are used in the training process, while the 61265 remaining ones are used for testing purposes of the probabilistic model.

Each frame is acquire and processed with the EM modeled naive Bayesian Classifier to obtain the current location as an input to the Kalman filter wich estimate the position of the data.

The first section of this paper will introduce the problem and characteristics of the used data, the second section introduce the mathematical models to the reader. The sections three and four shown the development and experimental results of the proposed method, while the last section presents the conclusions.

Mathematical models

This section introduced the mathematical expressions for the Kalman filter, the naive Bayesian classifier and the Expectation Maximization (EM) algorithm.

For each part of the final algorithm a few considerations are taken in order to achieve the results.

A. Kalman Filter

Kalman filter is a recursive algorithm wich estimates the position and uncertainty from a moving feature in the next iteration. It is supposed that the acquisition time from each image to be analyzed is $t_k = t_0 + k\Delta T$ for $k = 1, 2, ..., n$, whit a small ΔT value.

The system is modeled using the state vector s_k defined in (1) and a set of equations called the system model. The state vector contains the position (x_k , y_k) and velocity (v^x , v^y) of the object in the instant k.

$$s_k = [x_k \quad y_k \quad v_k^x \quad v_k^y]^T \tag{1}$$

The system model uses the expression in (2), where φk 1 is called the transition matrix that may or not be time dependent wich contain the relations between the past state with the current one.

$$s_k = \phi_{k-1} S_{k-1} + \xi_{k-1} \tag{2}$$

The relations in (3) are used to define ϕ_k ; these equations lead to (4).

$$\begin{aligned} x_k &= x_{k-1} + v^x_{k-1} \\ y_k &= y_{k-1} + v^y_{k-1} \\ v^x_k &= v^x_{k-1} \\ v^y_k &= v^y_{k-1} \end{aligned} \tag{3}$$

$$\begin{pmatrix} x_k \\ y_k \\ v^x_k \\ v^y_k \end{pmatrix} = \begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_{k-1} \\ y_{k-1} \\ v^x_{k-1} \\ v^y_{k-1} \end{pmatrix} + \begin{pmatrix} \xi_1 \\ \xi_2 \\ \xi_3 \\ \xi_4 \end{pmatrix} \tag{4}$$

The ξ variable represent the system perturbations due to additive noise and it is modeled as a gaussian white noise with mean zero and covariance matrix Qk

$$Q_k = \begin{pmatrix} \sigma^2_Q & 0 & 0 & 0 \\ 0 & \sigma^2_Q & 0 & 0 \\ 0 & 0 & \sigma^2_Q & 0 \\ 0 & 0 & 0 & \sigma^2_Q \end{pmatrix} \tag{5}$$

Since this filter estimate the state, it is supposed that a noisy measure is taken. This relation is shown in (6)

$$z_k = H_k s_k + \mu_k \tag{6}$$

Where zk is formed with the positions of the object measured with a sensor, but since a noisy measure is supposed, the µ component is added deriving in the following relation

$$\begin{pmatrix} z^x_k \\ z^y_k \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \begin{pmatrix} x_{k-1} \\ y_{k-1} \\ v^x_{k-1} \\ v^y_{k-1} \end{pmatrix} + \begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix} \tag{7}$$

Where µk defines a white gaussian noise with mean zero and covariance matrix Rk

$$R_k = \begin{pmatrix} \sigma^2_R & 0 \\ 0 & \sigma^2_R \end{pmatrix} \tag{8}$$

Once the system is modeled the Kalman equations are applied, these equations include the estimator precovariance (9), the Kalman gain (10), the optimal state estimator (11) and the estimator covariance matrix (12).

$$P'_k = \phi_{k-1} P_{k-1} \phi^T_{k-1} + Q_{k-1} \tag{9}$$

$$K_k = P'_k H^T_k (H_k P'_k H^T_k + R_k)^{-1} \tag{10}$$

$$\hat{s}_k = \phi_{k-1} \hat{s}_{k-1} + K_k (z_k - H_k \phi_{k-1} s_{k-1}) \tag{11}$$

$$P_k = (I - K_k H_k) P'_k \tag{12}$$

B. Bayesian classifier

Is a probabilistic classifier based on two assumptions:

The decision problem can be described in probabilistic terms.

The probability for all possible values is known.

This classifier uses the Bayes theorem stated in (13).

$$P(h|D) = \frac{P(D|h)P(h)}{P(D)} \tag{13}$$

Where

h correspond to the hypothesis D is the observed data P(h) and P(D) are the probabilities of h and D independent of each other.

P(Djh), a conditional probability, is the probability of the the observed data D once the h hypothesis is accomplished.

P(hjD), is the probability of occurrence of h hypothesis given the D data.

Since P(D) is considered a constant (for all the hypothesis), it can be omitted from the decision, and then we can use the maxima a posteriori (MAP) shown in (14) or the maximum likelihood (ML) shown in (15). The use of one of other will depend in the training mode, if each hypothesis has the same occurrence probability the ML classifier is used, otherwise the MAP classifier is used.

ISSN-On line: 2414-4924

$$h_{MAP} = \arg \max P(D|h)P(h) \tag{14}$$

$$h_{ML} = \arg \max P(D|h) \tag{15}$$

C. Expectation Maximization: GMM

The EM algorithm group a data set defining the probability of each one each group, this is based on an statistic model of finite mixture of gaussian functions.

This model describe a set of k groups with a gaussian or normal probability distribution for each one. The parameters of the distributions can be calculated using the probability of belong of each data and the data itself; the equations (16) to (18) describe the calculus for the mean, the variance and the probability of each group distribution respectively, in each case, the w variables correspond to the probability of the data to belong to the current group.

$$\mu_A = \frac{\sum_{i=1}^{n} w_{Ai} x_i}{\sum_{i=1}^{n} w_{Ai}} \tag{16}$$

$$\sigma_A^2 = \frac{\sum_{i=1}^{n} w_{Ai}(x_i - \mu_A)^2}{\sum_{i=1}^{n} w_{Ai}} \tag{17}$$

$$p_A = \frac{\sum_{i=1}^{n} w_{Ai}}{\sum_{i=1}^{n} w_{Ai} + \sum_{i=1}^{n} w_{Bi}} \tag{18}$$

With an aproximation of the parameters, we can calcule an estimation to the group applying the gaussian distribution defined in (19).

$$f(x;\mu_A,\sigma_A) = \frac{1}{\sqrt{2\pi\sigma_A^2}} e^{\frac{(x-\mu_A^2)}{2\sigma_A^2}} \tag{19}$$

In order to obtain a better estimation, the weights (or belong probability) are updated using (20) and the algorithm is repeated until a stop condition is accomplished.

$$w_{Ax} = \frac{f(x;\mu_A,\sigma_A)p_A}{f(x;\mu_A,\sigma_A)p_A + f(x;\mu_B,\sigma_B)p_B} \tag{20}$$

Since the EM algorithm only estimates the values and seldom achieve total convergence two common stop conditions are used: define a number of iterations and achieve a maximum likelihood for several iterations.

The first one can lead to a bad estimation if a small number of iterations is selected, while the second one can never be achieve; for these reasons they should be used together in order to avoid nonconvergence problems and obtain the better estimation. The maximum likelihood estimator can be used with any of the equations in (21).

$$LP = \prod_i p_A P(x_i|A) + p_B P(x_i|B)$$
$$LP = \sum_i ln(p_A P(x_i|A) + p_B P(x_i|B)) \tag{21}$$

Modeling the problem

In order to avoid a dependent component color space the sRGB images should be converted into another color space, the one we propose to use is the CIE-L*a*b color space. Since we do not require the colors of the segmented area but its current position we do not need to return the results to the sRGB color space.

Therefore the full process will consist in the segmentation of the skincolored area and the application of the Kalman filter. To achieve the segmentation, a naïve Bayesian classifier is used applying the EM algorithm to obtain the model. The system is modeled using the equations describe in section II.

A. The model problems

While histogram based models are non-iterative methods, the EM needs a stop condition in order to stop the loop since the total convergence is seldom obtained, the incorrect application of this condition can derive in misclassified data.

Since the output of the EM is only an estimation of the actual solution, the parameters for each class distribution function will show an estimate behavior as well.

The better the stop condition, the better the results; but sometimes the condition is achieve while the output is far away from the actual solution and therefore a poor modeled distribution will be obtain instead.

The Figure 1 show the class assign for each training sample once a certain stop condition is achieve for the training data set, this graph entail that while one class training data is correctly assigned the other one include misclassified points, leading to a poorly classified results each time this model is used. The stop condition to avoid classification problems should be stated depending on each particular problem needs.

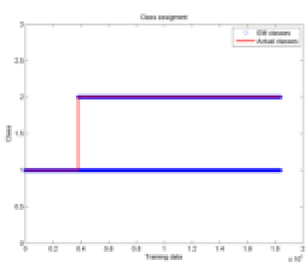

Figure 1 Convergence problem

There exist the possibility of multiple moving objects within the scene, but sometimes we will not need to track all of them; therefore we need an accurate method to detect and extract features from the desired objective.

Our selection of the Bayesian Classifier will reduce our problem to a probabilistic decision, but to choose the correct model is the real challenge.

Experimental Results

The data from Table 1 show the accuracy rate for several classifiers (Naive Bayes (Expectation Maximization with partial convergence (EM1), Expectation Maximization with total convergence (EM2) and histogrambased (Hist) pdf), Kmeans and KNN with pn rule) once they are used with CIE-L*a*b features (a component (a), b component (b), coste function (c), MatLab Gaussian fitting (f)) and the combination of them.

These results show that the use of EM algorithm outperform the classic histogram based model definition without the use of the coste function and the other two classifiers.

However, once the models are used with real data, not the one from the database, the EM models shown a better performance than the histogram based models since the first ones does not misclassify hair or clothes border pixels. In Figure 2, the versatility of the.

	Bayes			kmeans	KNN
Feature	EM1	EM2	Hist		
a	82	95.7	96.1	**93.5**	97
b	84. 2	91.7	96.7	91.9	97
ac	73.9	22.3	95.6	-	-
bc	**84.4**	22.2	96.7	-	-
ab	80.6	**98.9**	99.5	92.9	96.3
abc	79.3	23.9	**99.6**	-	-
af	82	95.7	94.3	-	-
bf	84.2	91.9	97.2	-	-
afc	73.9	22.3	21	-	-
bfc	**84.4**	22.4	53.2	-	-
abf	80.5	99	**96.9**	-	-
abfc	79.7	**99**	96.8	-	-

Table 1 Accuracy rate for different pdf models and attributes

Database and the models for segment different skin pigmentation and the problem generated for the EM convergence problem is shown.

ISSN-On line: 2414-4924

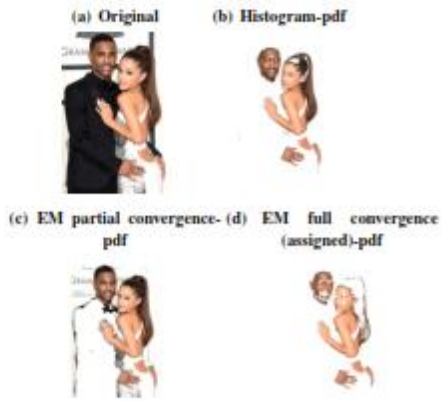

Figure 2 Skin-detection

With the features extracted, the Kalman filter can be applied. The Figure 4 contain the estimated path of a moving object once a noisy measure (current position plus white gaussian noise (N(0; 10))) is taken.

It is clear that even with a non-related initialization, the Kalman filter follow the actual path closely.

If the object is partial or total blocked the Kalman filter uses the estimation as observation and continue the process without problem.

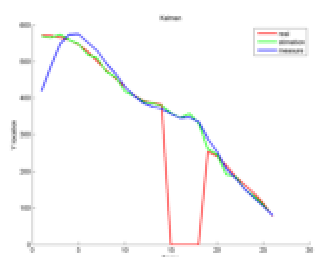

Figure 3 Kalman estimation of uncomplete data

To test the Kalman filter behavior an object is recorded on a video and its current positions based on the results of a EM modeled Bayesian classifier are obtained.

To simulate missing data the object cross behind a barrier, the results are show in Figure 4. The path in red marks the actual path of the object while the green one show the Kalman filter estimation. While we have not information of the object behind the barrier, the Kalman filter estimate its position and possible exit from it.

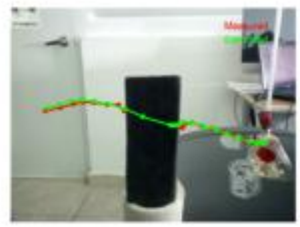

Figure 4 Kalman tracking of a total blocked object

Conclusion

EM showed a better performance than histogram based (hb) models, at least on natural images; while the hb models classify non-skin pixels with similar colors as skin ones, the use of EM assign them correctly as nonskin.

The application of the Kalman filter to found missing data showed a good performance as well, while some models consider a constant velocity, an approximate value can be found using the current and past location, solving the missing data problem and estimating a path that follows the behavior of the tracked object.

References

M. Lichman, "UCI machine learning repository,"
2013.