Human Detection and Motion Tracking

Technical report - FI - VG20102015006 - 2011 – 04

Ing. Ibrahim Nahhas

Ing. Filip Orság, Ph.D.

Faculty of Information Technology, Brno University of Technology

December 9, 2011

Abstract

This document is a compilation from three different articles focused at human and face detection, tracking and motion analysis. It is not meant to be an original work, it is merely a commented digest from the three published articles covering the topics of human detection and tracking in video sequences. It is not completely closed and finished version either, rather it is the first version summarizing results of search done until December 2011.

Content

1 Introduction

In this document we summarized a few published articles. We extracted information useful for our research and commented it for our internal purposes only. This document is not meant to be published and is not an original work.

Human detection and tracking is a difficult task. The difficulty lies mainly in the complexity, since people interact with each other, form groups and may move unpredictably. Generally, there are two main scenarios – either the camera is fixed or movable. Those two aspects, which seem to be irrelevant, are of high importance to choices we have to do while designing an automated detection and tracking system.

According to the two defined scenarios, we can say that either the background of the watched scene is static or not. In this document we will focus on the fixed camera scenarios, since it is an easier choice to start with.

To get over all the challenges in a complex environment using a fixed camera to detect people and track their movements, we have to design a robust background model, which can deal with slow illumination changes (day and night) and fast illumination changes (clouds blocking the sun).

The following chapters are digests of a few articles and many parts of them are simply fully cited with some comments when needed. The cited paragraphs are differentiated by an italic text.

2 Real-time human motion detection and tracking

This chapter is based on an IEEE article: ZARKA, Nizar, Ziad ALHALAH and Rada DEEB. Real-Time Human Motion Detection and Tracking. In: *2008 3rd International Conference on Information and Communication Technologies*: *from Theory to Applications: Damascus, Syria 7-11 April 2008*. Piscataway, NJ: IEEE, c2008, pp. 1-6. ISBN 978-1-4244-1751-3. DOI: 10.1109/ICTTA.2008.4530098.

Detection and tracking are achieved by several steps:

- create a background model that can deal with lightning changes, long term changes in the scene and objects occlusions
- get foreground pixels by using the background subtraction method
- noise cleaning and object detection
- human modeling to recognize and monitor human activity in the scene such as human walking or running

There are some vision systems [i.e. Pfinder (Wren, C. et al.: Pfinder: Real-Time Tracking of the Human Body. *In: IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1997), W4 (Haritaoglu, I. et al.: W4: Who? When? Where? What? A Real Time System for Detecting and Tracking People. Computer Vision Laboratory, University of Maryland College Park. 1998)] for people detection and tracking using features like head or body shape, and statistical models restricting them to human-oriented applications. Large number of pixels need is solved by a star skeleton model main idea in this model is a simple form of skeletonization, which only extracts some internal motion features.

2.1 Background modeling

The background is the image which contains the non-moving objects in a video and the background model is done in two steps: first, the background initialization, where we obtain the background image from a specific time from the video sequence, then, the background maintenance, where the background is updated due to the changes that may occur in the real scene.

2.1.1 Background model initialization

The median algorithm is widely used in background model initialization. It is based on the assumption that the background at every pixel must be visible more than fifty percent of the time during the training sequence. The median algorithm results in wrong background intensity value, especially when a moving object stops for more than fifty percent of the training sequence time.

More efficient algorithm is the Highest Redundancy Ratio (HRR) algorithm which considers that pixel intensity belongs to the background image only if it has the highest redundancy ratio among intensity values of the pixel taken from

a training sequence. This assumption is very near to the actual meaning of the background. Therefore, HRR algorithm is more flexible and applicable for real events than median algorithm.

2.1.2 Background model maintenance

In a scene, many changes can occur and may impact the background image. These changes are i.e. illumination changes (sun location, cloudy and sunny weather, turning the light on/off etc), position/motion changes (small camera displacement, tree branches moving etc.), and changes introduced to the background (originally moving objects in the scene which stay without moving for a long time). These changes must be tolerated by the background model.

There are two types of adaptations: sequential and periodical. The sequential adaptation can be done by using a statistical background model = i.e. *low pass filter* applied on each pixel.

The periodical adaptation is used to adapt to high illumination and physical changes that may happen in the scene, like deposited or removed objects. In this adaptation the background model is re-initialized using the HRR algorithm and this is done in a periodical sequence (50-100 frames).

2.2 Foreground and feature extraction

Foreground, for our purposes, is an object being tracked = an unknown object (part of the foreground) in the scene (background). In contrast to the background, the foreground (or the object) cannot be easily adopted to. To separate the foreground from the background in order to detect the moving objects, we can use image segmentation.

For the purposes of the foreground region extraction, we can use simple background subtraction method. An updated background image over time is subtracted from the current frame. Result of the subtraction (pixels with a high value of intensity) are flagged as foreground pixels and considered background pixels otherwise.

A statistical analysis of background pixel intensity shows that its value can be modeled using Gaussian distribution. So if a difference pixel has a value more than three times the standard deviation 3σ for this pixel value from the background pixel intensity, then it is marked as foreground pixel.

Value of 3σ is usually considered a good value. A 3x3 median filter is then used to reduce misclassification.

Determining of an accurate position of objects in the foreground image is the very basic task of a tracking system. In many applications is the 2D image represented by its 1D projection histogram over the horizontal and the vertical axis. In order to distinguish among many objects, we can use the same algorithm for each bounding box (hence for each object being tracked). After getting the bounding boxes for the objects, a simple filtration should take

place to eliminate small objects, considering them as irrelevant or misclassified objects.

Foreground region extraction and object detection process may produce images that contain holes and gaps in the object detected. So we need to restore the objects to their original state and size by applying a sequence of two dilations and one erosion. The contour is then detected using Duda & Hart algorithm.

Robust tracking can't be implemented without good features. The features used in the source paper are the following ones:

- *The center of gravity (centroid), which is used as a representative of the object to detect its location in the frame sequence.*
- *The velocity, which is calculated in each frame to help predicting the next position of the object and its moving direction.*
- *The object size is number of foreground pixel the object has.*
- *The color helps the tracking algorithm to make use of appearance information of the objects to separate them in case of occlusion in which the previous features are vague and difficult to calculate.*

2.3 Object tracking

The tracking starts when a new object enters the scene. The following steps are performed:

- *Compute the correspondence between the new object and the objects currently being tracked.*
- *Use tracking algorithms to estimate the position of each object ,which has to achieve high precision moving object correspondence in presence of noisy foreground image, changes in the size of regions, occlusions and entering/exiting of objects.*

Motion correspondence deals with regions defined:

- *The 2D coordinates of the centroid (**P**)*
- *The bounding box **(B)***
- *The size **(S)***
- *The color and the gray level characteristic **(D)***
- *The velocity **(V)***
- *The predicted change in size* ($\boldsymbol{\nabla S}$)**.**

In frame [t] of the sequence we define:

***New regions**, a group, of the size N contains the regions with centroids P_i^t (where i is the region number $0 \le i \le N$) whose correspondences to previous frame are unknown. **Old regions**, a group, of the size M contains the regions with centroids P_L^{t-1} , where L is a label, whose correspondences to previous frames have been established.*

The number of regions in frame [t] might be less than the number of regions in frame [t-1] due to exits or occlusion and it can be larger due to entries or splitting.

Some regions may have not a corresponding object. These regions are involved when an object is entering/exiting the scene, or during object occlusions, which happens when two objects or more are close to each other so the background subtraction method gives a single region for them.

Such situation must be dealt with. There are number of solutions. The paper presented in this chapter detects three typical situations and reacts accordingly. Those situations are:

- Detection of objects exiting the scene
- Detection of objects entering the scene
- Detection of occlusion

For each remaining regions in old regions group, if the estimated centroid position is outside the scene, then the region is considered an exiting region.

For each remaining regions in new regions group, if one of the lines forming the bounding box overlaps one of the scene borders, then the region is considered an entering region, and it is marked with the appropriate label, and it is added to the matched regions group and deleted from the new regions group.

If there is a region from the old regions group its predicted bounding box intersect with the new region bounding box, we consider that the old object is occluded by the new one.

2.4 Human modeling

In order to correctly recognize humans in the scene we can use model of a human to be able to recognize them. There are many ways to solve this. However, in the presented paper there is mentioned method based on skeletonization. That approach to the problem is called *Star Skeletonization*, based on the five most significant parts of the human body (head, arms, and legs).

Such model has been used in many tracking systems. Skeletonization itself can be performed in many ways as transformations. Those cost much time, though, hence are not perfect for real-time applications. In the proposed solution, there is a star skeleton construction method defined. This method is less demanding to the hardware and is proved to be robust when dealing with noise. This is achieved by applying a few steps of abstraction on the objects (boundary extraction, location of the human body = detection of the extremes) and skeleton configuration.

The location of the human body (extremal points) is defined by unwrapping the bounding contour into an Euclidean distance signal that represents the distance between the centroid of the object and the point in the object's contour.

Next a median filter is applied to smooth the distance signal. Then the local maxima are extracted by monitoring the changes in the amplitude values of the distance signal, by detecting the zero crossing. Building the star skeleton is now easy by connecting each of the extracted local maximum to the centroid of the object.

There are many advantages offered by this type of skeletonization such as scalability, computationally cheap, size insensitivity and it is applicable for different kind of objects especially to those which exhibits periodic movement and not only humans.

2.5 Experimental results

Results of experiments based on the above described method were summarized as follows:

For the indoor scenes the tracking is achieved with an accuracy of 96%, 86%, 67% for one, two and three people respectively in the scene. For the outdoor scenes the tracking is achieved with an accuracy of 87%, 83%, and 64% for one, two and three persons respectively in the scene.

The numbers presented here have been well expected. It is clear, that for the outdoor applications we have to find a better solution. For the indoor environment we can use the proposed solution. However, finding a better solution for the outdoor application means it could perform even better indoors, which would make the described method unnecessary.

3 A human model for detecting people in video from low level features

In this chapter we summarized results presented in HARASSE, S., BONNAUD, L., and DESVIGNES, M. A Human Model for Detecting People in Video from Low Level Features. *In: 2006 IEEE International Conference on Image Processing: Atlanta, USA 8-11 October 2006*. IEEE, 2006, pp.1845-1848. ISBN 1-4244-0480-0, DOI: 10.1109/ICIP.2006.312839. In this article the authors aim at detecting people in video streams using skin color and foreground probability maps by defining the spatial relationship between those. A new term is introduced – human signature – which represent a human detected in the video stream.

Similarly to the article presented in the previous chapter, in the approach presented here the algorithm must distinguish between foreground and background too. Since the methods are similar we will proceed right to the human description and model.

3.1 Human model

The model is based on spatial regions. Exact explanation follows.

The model describes the appearance of the human body in three spatial regions. Those are the head region with foreground pixels, the face region with skin pixels, and the body region with foreground pixels. Pixels surrounding the head and face regions must be background and non-skin respectively.

By defining a different region for the head more than for the rest of the body, most occlusion issues are avoided, because heads are well separated one from each other in the considered context. Indeed, two people whose bodies appear as a single cluster of foreground pixels can be detected as two different people.

Practically, the model is represented as a vector containing the 4 parameters of 3 rectangles, one for each region. It permits a low dimensionality of the vector, as well as a fast computation of the parameters.

For the purposes of the model estimation, a Bayesian framework is used. The model vectors are based on probabilities of a vector being similar enough to be the modeled human. Hence, this approach is based on probabilistic modeling. As such, estimation of the parameters is performed basically by a Monte Carlo simulation and later on it is corrected to minimize likelihood of the vector representing the human being tracked.

3.2 Human appearance signature

Taking in account the above algorithm will be used for modeling people, we have to deal somehow with more than one person being present in the scene. Hence, we assume more occlusions will occur, which means the tracking

algorithm might lose the targets frequently if it was not capable of identifying each person in the image. The modifying nature of the human body (hands, legs) makes this task non-trivial. For this task a human appearance signature is used. The algorithm is described in the original paper as follows:

Color appearance

The shape of a person has too much variation during the video sequences to be useful for tracking. On the other hand, the color of clothes is stable enough to be matched between consecutive frames. The proposed signature consists in a small number representative of the general appearance of the considered person. The two steps of the algorithm are the computation of the signature and the definition of a distance between signatures to achieve matching.

Computation of the color signature

A color signature is computed from the body rectangle as shown figure 12 in each detected person. Pixels in the body region include pixels from the actual clothes of the person, but it may also contain skin pixels, background pixels and pixels from other people, when the human model does not perfectly match the body. Only the colors of clothes from the considered human body are interesting for the signature. The pixels in the body are quantized into 5 colors (assumed to be sufficient to represent all colors of a person's clothes) with a k-means clustering algorithm. The color signature S_n *for the considered person n, noted as* $S_n = \{ C_1^n, C_2^n, C_3^n, C_4^n, C_5^n \}$.

Matching signatures

The matching of signatures between two frames permits the association of detected people in a frame to detected people in another frame. It is necessary to define a distance between signatures. This distance must be robust to colors in the signatures that originate from other objects than the considered person. First, a distance between a color c and a signature S_n *is defined as* $D(c, S_n) = \min(\|(c - c_i^n)\|, i = 1... 5)$ *. It is assumed that two signatures can be matched when at least 3 colors are similar.*

3.3 Experimental results

Results provided by this approach according to the paper prove this method to be very promising. In the analyzed videos, 97% people were detected correctly, while 4% were false detections (the sum of both is not 100%, since there were some misdetections present too).

4 Human face detection using skin color information

In this chapter we summarized results presented in SHEHADEH, H., AL-KHALAF, A., AL-KHASSAWENEH, M. Human Face Detection Using Skin Color Information. *In: 2010 IEEE International Conference on Electro/Information Technology (EIT): Illinois, USA 20-22 May 2010*. IEEE, 2010, pp. 1-5. ISBN 978-1-4244-6873-7, DOI: 10.1109/EIT.2010.5612128.

To identify persons and distinguish them in a complex environment, we have to employ an approach to detect human faces in color images. For this purpose we can use two natural and powerful factors – skin color and eyes. Method presented in the cited paper takes two steps in order to extract regions of the human skin color and the eyes in the images.

In the first step of the processing, those color regions, which probably are human skin color, are extracted based on the detection of all pixels in colored images. For this purpose a reference skin colors matrix is used. One of the most important features (except for color, obviously) is circularity of the face. Hence, the region has to be roughly circular in shape to be a human face. Those regions are marked as candidates for being human face.

In the second step, the eyes are extracted from the candidate regions by calculating central moments and by using geometrical relationships between the eyes. The position and angles between the eyes can determine the exact location of the face.

4.1 Skin color pixels

Skin color detection can be performed in different color spaces. Even a combination of different color spaces can be taken into account to get more accuracy in skin and face detection.

The main challenge at this stage is to train the computer to all possible skin colors. Then, having a set of allowed colors, the computer is able to select only those pixels of an input image that represent the color of the body. Then other facial characteristics (such as geometry and any other properties) must be used to distinguish and localize the faces from other parts.

The main idea is to create a database containing all the pixels that represent the colors of the human skin.

The problem is that the color of the human body varies from person to person, and we have different colors for different races. Therefore, the pixels compilation process that represents the human skin must compile pixels from different images and different skin colors.

4.2 Skin detection

For the purposes of the face detection, in the presented paper a color matrix is used. The matrix contains a collection of RGB values representing skin color scale. Pixels in an RGB image consist of three values – red, green and blue.

Since the reference matrix used contains all RGB vectors that might be a skin color, then the system computes the difference (norm) between each vector in the reference matrix and pixels in the input RGB image within a loop, the equation used to calculate the difference is:

$$norm = \sqrt{(R_i - R_f)^2 + (G_i - G_f)^2 + (B_i - B_f)^2},$$

where

- R_i*: The Red component in the input image.*
- R_f*: The Red component in the reference matrix.*
- G_i*: The Green component in the input image.*
- G_f*: The Green component in the reference matrix.*
- B_i*: The Blue component in the input image.*
- B_f*: The Blue component in the reference matrix.*

Once the difference is less than a threshold, this pixel is set to be a skin. The results of this process is a binary image contains 1's and 0's; while the 1's represent all pixels that have similar skin color, and the 0's represents non-skin pixels.

Result of this process is a binary image with areas set to 1 in case of pixels being part of the face. Not all areas belong to the face, though. A post-processing must be applied to the image to remove irrelevant areas. The first step is to remove areas too small to be face. Drawback of this is possibility the result will be not a smooth, flat surface – it might contain holes inside, which can be fixed too. Last part of this process is application of edge detection. Background pixels with values close to the skin color values would be false recognized as skin. For this purpose a simple edge detection algorithm helps to separate actual skin regions and background.

4.3 Face detection

With binary image on hand we can detect face in it. For this purpose, there are many algorithms available. Those rely on various features provided by faces. In the presented article the face detection is performed as follows.

Once a connected component analysis is applied on the modified skin binary image, we can get the number of objects and the binary labeled matrix. Any features for all objects can be calculated. Our interests are the area and the perimeter for all objects. The area of each object is defined as follow:

$$area = \sum_{r,c \in R} 1$$

where r and c are the row and column of image R, respectively. It is the summation of all the 1's in each connected object. The perimeter is the calculated by this equation:

$$perimeter = \#\{k|(r_{k+1}, c_{k+1}) \in N_4(r_k, c_k)\} + \sqrt{2}\#\{k|(r_{k+1}, c_{k+1}) \in N_8(r_k, c_k) - N_4(r_k, c_k)\}$$

where k+1 is the modulo K; N4 is the 4-connected neighbors and N8 is the 8-connected neighbors. The circularity of each object can be calculated as follow:

$$circularity = \frac{|perimeter|^2}{area}$$

The circularity describes how circular the shape of an object is. If this value is small, then the object will look like a circle. This feature was chosen since the human face is likely to be circular. A threshold value for the circularity was set to work on different images such that the system will locate the faces and bound them in boxes.

4.4 Experimental results

According to the presented article the algorithm was tested on many images. Exact numbers representing the results were not presented (*simulations give satisfactory results*). According to the Figure 1 below, the detection works well enough for further processing purposes (i.e. people identification). Advantage of this algorithm is its ability to recognize faces of people of various races.

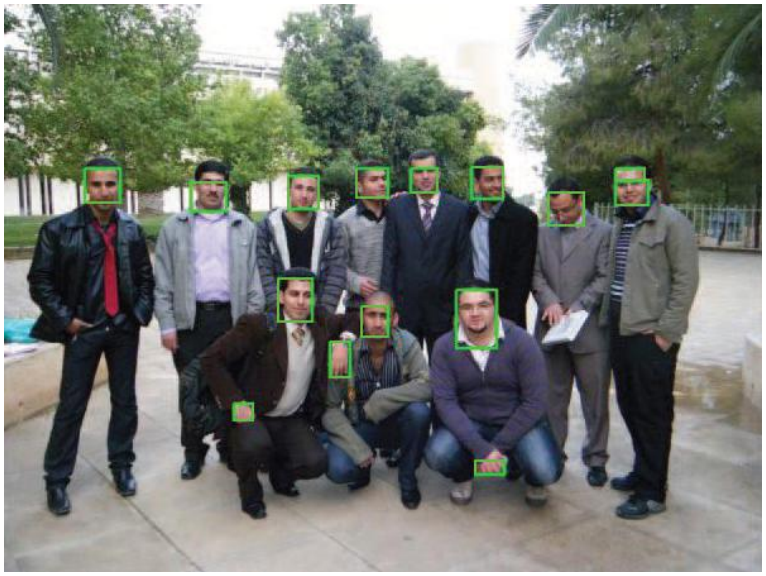

Figure 1. Faces detected by the presented algorithms.

5 Conclusion

This document summarizes some of to date algorithms and their results in the field of human detection and face detection in images. This document is not a finished work. It is merely a sum-up from the first searches for the current algorithms available in public libraries. As work on this topic will go on in the following year, this document will grow. It will be reorganized, filled with new information and modified to our needs with respect to the MVCR project, part of which this document is.

6 References

[wren97] WREN, C. et al. Pfinder: Real-Time Tracking of the Human Body. *In: IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1997.

[hari98] HARITAOGLU, I. et al. W4: Who? When? Where? What? A Real Time System for Detecting and Tracking People. Computer Vision Laboratory, University of Maryland College Park. 1998.

[zark08] ZARKA, N., ALHALAH, Z., and DEEB, R. Real-Time Human Motion Detection and Tracking. In: *2008 3rd International Conference on Information and Communication Technologies*: *from Theory to Applications: Damascus, Syria 7-11 April 2008*. Piscataway, NJ: IEEE, c2008, pp. 1-6. ISBN 978-1-4244-1751-3. DOI: 10.1109/ICTTA.2008.4530098.

[hara06] HARASSE, S., BONNAUD, L., and DESVIGNES, M. A Human Model for Detecting People in Video from Low Level Features. *In: 2006 IEEE International Conference on Image Processing: Atlanta, USA 8-11 October 2006*. IEEE, 2006, pp.1845-1848. ISBN 1-4244-0480-0, DOI: 10.1109/ICIP.2006.312839.

[sheh10] SHEHADEH, H., AL-KHALAF, A., AL-KHASSAWENEH, M. Human Face Detection Using Skin Color Information. *In: 2010 IEEE International Conference on Electro/Information Technology (EIT): Illinois, USA 20-22 May 2010*. IEEE, 2010, pp. 1-5. ISBN 978-1-4244-6873-7, DOI: 10.1109/EIT.2010.5612128.