



ISSN: 0067-2904

Real Time Multi Face Blurring on Uncontrolled Environment based on Color Space algorithm

Alya'a R. Ali*, Ban N. Dhannoon

Department of Computer Science, College of science, Al-Nahrain University, Baghdad, Iraq

Abstract

Faces blurring is one of the important complex processes that is considered one of the advanced computer vision fields. The face blurring processes generally have two main steps to be done. The first step has detected the faces that appear in the frames while the second step is tracking the detected faces which based on the information extracted during the detection step. In the proposed method, an image is captured by the camera in real time, then the Viola Jones algorithm used for the purpose of detecting multiple faces in the captured image and for the purpose of reducing the time consumed to handle the entire captured image, the image background is removed and only the motion areas are processed. After detecting the faces, the Color-Space algorithm is used to tracks the detected faces depending on the color of the face and to check the differences between the faces the Template Matching algorithm was used to reduce the processes time. Finally, the detected faces as well as the faces that were tracked based on their color were obscured by the use of the Gaussian filter. The achieved accuracy for a single face and dynamic background are about 82.8% and 76.3% respectively.

Keywords: Viola-Jones algorithm, Face detection, Face tracking, Color space, Real time.

تغويش عدة وجوه في بيئه غير منضبطه في الوقت الحقيقي

علياء رفعت علي*، بان نديم ذنون

قسم علوم الحاسوب، كلية العلوم، جامعة النهرين، بغداد، العراق

الخلاصة

تعد عملية تغويش الوجوه إحدى العمليات المعقدة والهامة التي تعتبر واحدة من حقول الرؤية الحاسوبية المتقدمة. عادةً، ما تمر عمليات تغويش الوجوه بخطوتين رئيسيتين يجب القيام بهما. الخطوة الأولى في الكشف عن وجود الوجوه التي تظهر في الصور بينما تكون الخطوة الثانية هي تتبع الوجوه المكتشفة والتي تستند إلى المعلومات التي تم استخراجها أثناء مرحلة الكشف. في الطريقة المقترحة، يتم التقاط الصورة بواسطة الكاميرا في الوقت الحقيقي، حيث تم استخدام خوارزمية فيولا جونز لغرض اكتشاف الوجوه متعددة في الصورة الملتقطة، ولغرض تقليل الوقت المستهلك للتعامل مع الصورة الملتقطة، تتم إزالة خلفية الصورة ويتم معالجة مناطق الحركة فقط. بعد اكتمال عملية اكتشاف الوجوه، يتم استخدام الخوارزمية المستندة إلى اللون (Color Space algorithm) لتتبع الوجوه المكتشفة بناءً على لون الوجه و للتحقق من الاختلافات بين الوجوه تم استخدام خوارزمية مطابقة القالب (Template Matching) لتقليل وقت المعالجة.

*Email: alyaa_iraq2005@yahoo.com

وأخيراً ، تم تغويز الوجوه المكتشفة وكذلك الوجوه التي تم تتبعها استناداً إلى لونها باستخدام Gaussian Filter). بالإضافة لذلك النتائج المحققة للخلفية الفردية والديناميكية هي حوالي 82.8% و 76.3%.

1. Introduction

In recent years, the researches that contain human face in the field of image processing have been grown in terms of interest, because of the establishment and development of some approaches such perceptual user interfaces and security applications, compression, and some others . Also in an image processing system, these approaches can blur, detect, track or recognize a face or an object in an image or in a series of images [1]. Human face detection and tracking system have been importantly developed and used in such PC cameras, digital monitoring, notebook, 3G cell phones, intelligent robots, digital cameras, and the like. Such that, these applications play an important role in our day to day life [2]. Object detection mechanism always the first requirement for every tracking object method, it's either in every frame or when the object first appears in the videos which it cannot be tracked the object without detected it [3]. In 2010, Niazi and Jafari, propose a Hybrid face detection algorithm that could detect faces in color images with different complex backgrounds and lights. Their method first detects face regions using HAAR classifier over an entire image and generate candidates for next classifier. HAAR classifier, usually detect all the faces in the image but also miss classified some none-face object as face. after that, by comparing RGB and HSV method they come into the result that HSV method has a better performance than RGB method by using simple feature based method named HSV color model to eliminate miss classified none-faces[4]. In 2013, Makovetsky and Petrosyan, presented a "Face Detection and Tracking with Web Camera", they use Viola-Jones algorithm for detecting the faces. Furthermore they use pixel sampling method to make this algorithm work faster by reduced the image size. Their system processes work on 8 frames per second and it is enabled to detect multiple faces and tracking all of them in real time. For the in-plane rotations, the algorithm can cover 45 degrees for left or right [5]. In 2016, Hendi and Mohammed introduce robot system for tracking moving objects, it uses color based tracking algorithm and border following algorithm to detect the location of the target object in the images. The proposed software robotic system succeeded to track the target object with a success rate up to 97% in control environment [6]. In 2017, Yee write a thesis titled "Cascaded Facial Detection Algorithms to Improve Recognition", it compares between three different types of facial detection algorithms Viola-jones algorithm, histogram of gradient algorithm, skin segmentation which combined in various configuration to test all the accuracy for the detection faces at the execution time then giving it into a convolution neural network which make it easy to identifying who is that person. Where the average time took to execute the Viola-Jones algorithm is 26.2 (ms) with FEI face database (Brazilian database of images) and the accuracy is 97.4% plus the error rate is 2.6% [7]. In 2018, Al-Mukhtar, proposed a paper titled "Tracking and Blurring the Face in a Video File" shows a detection, tracking and blurring a single face in video frame by using the Viola-Jones algorithm for detection the face and kanade-lucas-tomasi (KIT) feature tracker algorithm to tracks a set of feature points across the video frames . The results obtained that can be useful to blurring the face [8]. This research aims to protect the privacy by blur multiple faces in real time using Gaussian noise for blurring, Viola-Jones for detection and Color space for tracking. Figure-1 shows the block diagram of the system.

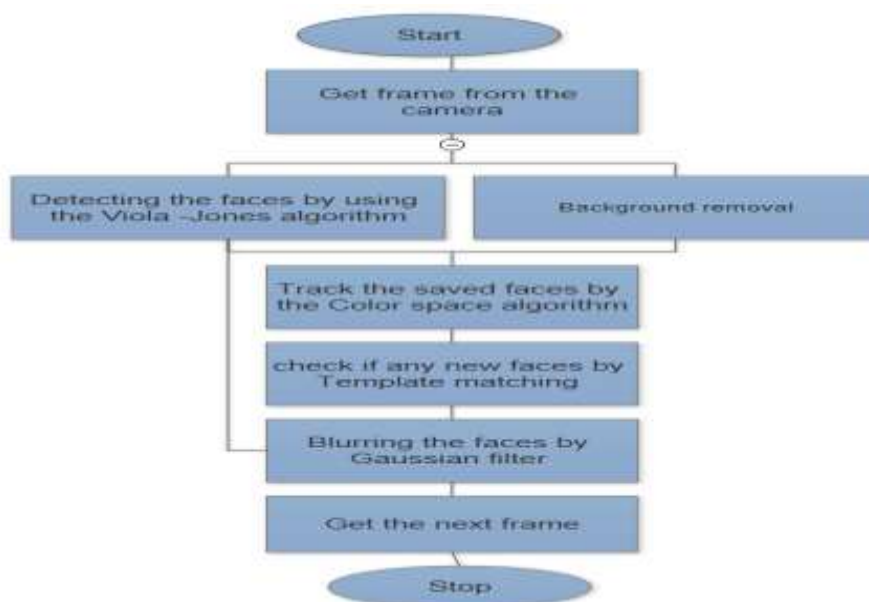


Figure 1-The block diagram of the system.

2. Template Matching

Template matching technique is a key component in many computer vision applications such as object tracking, detection, surveillance, and medical imaging. Furthermore, this technique is useful for searching and finding the location of a template image which (Small part of image) in the larger image. It is used in digital images processing to finding small parts of an image which matches a template image. It is used to find the pixel level and matches the character boundaries in the template image [9]. Usually, it is important because of the similarity index is used for comparing the template with the input image. Where the similarity index is determined on the following basis: It does not have to be sensitive to the noise of the image. It must be insensitive to changing image luminance. Its calculation burden must be light [10].

The “matching error” between the patch and any given location inside the image where this is being searched can be computed using different methods which are the square difference matching methods, Correlation coefficient matching methods, Normalized methods and Correlation matching methods. These methods multiplicatively match the template against the image means that a perfect match would be the largest [11]. Equation (1) shows its mathematical expression:

$$R(x,y) = \sum_{x',y'} (T(x',y') \cdot I(x+x', y+y')) \quad (1)$$

Where, I denotes the input image, T the template, and R the result [11].

3. The Proposed System

In the following sections the proposed system steps will be described in details:

3.1 Get The Frame From the Camera

The first step in the system is capturing a frame (represented as an image) in a real-time from the camera of the computer. After that, the captured frames will be displayed on the computer screen. These frames are the basis for the next steps of the system. The captured frames dimensions are (640 ×480) pixels for width and height, respectively.

The number of captured frames per second depends on the specifications of the camera, the number of the processed frames per second is 10fps.

3.2 Background Removing

Real-time systems always seek to reduce processing time for reliable results, video is the media that requires large resources to be processed. To minimize the number of resources used, avoid any additional unnecessary processes or operations. Bitwise XOR on images has been used to get the difference between two successive frames which the results of this step are a binary image include a difference between the two successive frames. The result of this step is a binary image include a difference between the two successive frames, but this image contains two types of noise described in the following paragraphs:

Probably, some pixels of the background connected to the foreground area of the image, to handle this problem, the **erosion** function was used in (6×6) black size by try and error in order to detect actual move and avoid simplest moves where the erosion process removes isolated foreground pixels. Then another problem arises due to the use of the erosion function, which is the presence of gaps inside the foreground of the image. To fill the small holes, the **dilation** function was used in (10×10) black size by try and error in order to expand white areas to illustrate it, the result is a binary image which then used the border following algorithm to finding the contours and labeling the motion regions. Figure-2 illustrates the example of XOR.

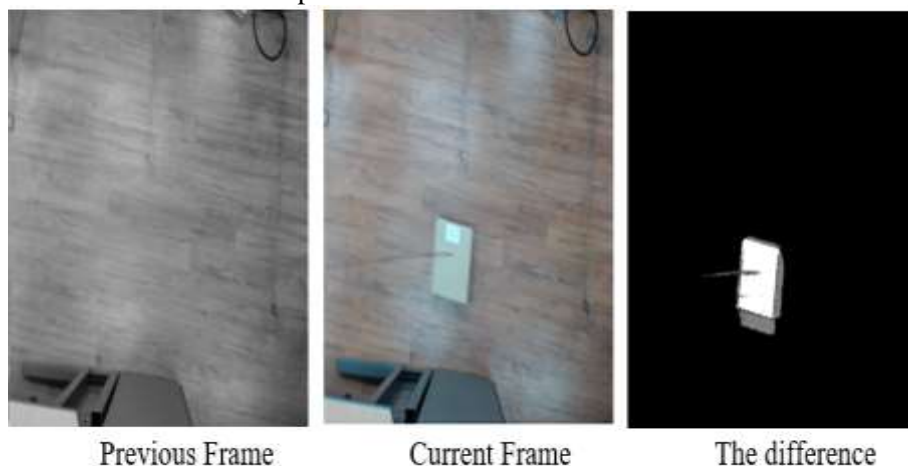


Figure 2-Example of the effect and output of implementation of XOR.

3.3 Face Detection Algorithm

Images taken from the camera may not be fully suitable for the face detection steps of processing. Gamma-correction must be taken before proceeding with basic processing to improve the image and give better and faster results. The value of Gamma-Correction used in this research is 0.5. A gray image is constructed of different shades of gray color. A true color image can be converted to a gray scale image by maintaining the luminance (brightness) of the image. The Table-1 below tastes and shows the result of the detection of a single face in a dynamic environment (uncontrolled background) for 10 frames per second.

Table 1-single face detection in uncontrolled background for 10 frame per second

Environment	Profile face			Total Time (ms)	Accuracy%
	Frontal Face	Left	Right		
	Time (ms)	Time (ms)	Time (ms)		
Uncontrolled	41.1	47.5	50.2	138.8	100%

- **Conversion of RGB image to grayscale image**

A gray-scale image is constructed of different shades of a gray color. A true color image can be converted to a gray scale image by maintaining the luminance (brightness) of the image. The RGB image converted to YUV but the Y value just calculated which is called the weighted average method the Y value calculated as the equation (2):

$$Y = 0.299 R + 0.587 G + 0.114B \quad \dots (2)$$

3.3.1 Viola-Jones algorithm

Viola-Jones is an algorithm used for detection the faces that appear in front of the web camera. It accomplishes detection by performing the following four steps:

- Haar like Feature to detects the areas of interest.
- Integral image to speeds up this process.
- Adaptive Boosting.
- Cascading classifiers.

The Haar-like features used in the Viola-Jones algorithm, that is, a scalar product between the image and some Haar templates. Equation (3) is to extract the feature:

$$V = \sum(P_b) - \sum(P_w) \quad \dots (3)$$

Where V is the value, P_b is a pixel from the black area, P_w is a pixels from black area. This algorithm looks for a specific hair feature of a face if these features found, algorithm passes the candidate to the next stage, Here the candidate are not whole image but just rectangular parts of this image known as sub window have a size of 20×20 pixel. With this window, the algorithm will check the whole image.

In this research, the Viola-Jones algorithm was used to detect the faces in two cases of faces:

- **Front face detection**

Detection faces that are directly in front of the camera and even faces that turn towards one of the parties in a simple way can be detected in this case.

- **Profile face detection**

In this case, faces that turn to either direction can be detected and divided into two by the direction they turn to:

- Turn to the left.
- Turn to the right.

To detect the faces that turned right there is a need to rotate the image with 180 degree to detect the faces in the image because of the Viola-Jones detection algorithm is designed to detect the faces that turned to the left side only

3.4 Find Mean HSV Pixel

For each face currently detected by the Viola Jones algorithm, one color value representing this face will be calculated as follows:

1- Convert each detected face from RGB color space to HSV color space. Pseudo code converting image from RGB to HSV.

RGB to HSV
$V = \text{Maximum}(R, G, B)$
$\Delta = V - \text{Minimum}(R, G, B)$
$S = \text{choose}(V=0, 0, \Delta * 255 / V)$
if $S = 0$
$H = 0$
else
if $\text{Red} = V$
$H_0 = (\text{Green} - \text{Blue}) / \Delta$
Else if $\text{Green} = V$
$H_0 = 2 + (\text{Blue} - \text{Red}) / \Delta$
else
$H_0 = 4 + (\text{Red} - \text{Green}) / \Delta$
end If
$H_0 < 0$
$H_0 += 6$
end
$H = H_0 * 255/6$
End

2- Find the arithmetic mean of the HSV pixels that make up the face image. Then stores this arithmetic mean for the purpose of benefiting from it in the process of comparison and search for the face in the next steps.

3.5 Tracking Object

This part of the system is interested in tracking the faces that were detected in the previous steps. Track the faces based on the color of faces and this operation depends on the analysis of images that captured from the camera. The color values of each face detected in the face detection part are summarized into a single color value that reflects the color values of the detected face. By looking for these color values in the image to be processed, some areas of this image with similar color values with these color values will be found. These areas will represent the faces that will be tracked.

Before proceeding with the tracking process, a difference will be checked between the current and the previous frames using the background removal method described in section 3.2. If there is no difference between the current and the previous frame, the operations will be stopped directly and returned to capture a new frame from the camera. Tracking part is divided into two sections depending on the number of the steps and presence of faces that detected in the previous steps or not

i. Initial step

If there are no detected faces from the current faces detection step and there is no color values were stored previously for the purpose of tracking them, then there is no need to complete the remaining steps of the algorithm and currently, it must be returned to get a new frame from the camera and start processing again. In case of detecting any face as a result of the previous face detection step, this means is the first time that the faces have been detected. All faces that are currently detected will be considered as a new faces and their color values will be stored for the purpose of tracking. The faces that are currently detected are the faces that need to be tracked because there are no faces detected before them.

ii. Tracking step

Lack of detection of any faces as a result of the current faces detection step means there is no need to track any new faces. Here the tracking process will be neglected and the algorithm will jump to the faces blurring step in the algorithm immediately and will blur the faces depending on the color values that were stored previously to be tracking them. In the case of any face detection in the previous steps, there is a need to know if this is the first time of detected a face or not. Checking must be done to know whether the current detected face is new faces or not. If this face is previously detected and processed, that means there is no a need to process and add it to the faces to be tracked again.

Each currently detected face must be compared with all the faces stored in the target faces array that is to be tracked by using Template Matching Algorithm to determine whether the face is new.

3.5.1 Check New Faces

In order to minimize the time and effort to tracking and blurring faces, duplicate faces that have been detected previously are ignored and only new and non-duplicate faces are tracked and blurred. In this research, the Template Matching algorithm was used to examine the faces and check whether they were new or not. The technique of template matching is used for finding area of an image that match the original to template image .

Result =TemplateMatching (Input source image, input template image)

where

Source image: Image where the search is running.

Template image: Searched template.

Result: A logical variable determines whether there is a similarity or not between the image and the template.

To identify the matching area, there is a need to compare the template image against the sources image by sliding it, which means moving the patch one pixel at a time (left to right, up to down). At each location, a metric is calculated so it represents how “good” or “bad” the match at that location is (or how similar the patch is to that particular area of the source image).

3.5.2 Identification of candidate areas to be faces

These specific areas were nominated based on the number of tracking and comparisons operations with the previously identified faces. These operations will be applied to each frame being processed after capturing from the camera. The following processes is a detailed explanation of each operation:

i. Down-sampling

The resolution of the captured image is 640 ×480 pixel where this resolution is high in order to achieve the process in the appropriate time. Therefore, the resolution of the captured image is down sampled using the down sampling steps of the Gaussian construction .

- Convolve the source image with the 5×5 Gaussian pyramid kernel.

- By rejecting even rows and columns in Downsamples the image. In the current research, the image is downsampled twice and the size of the output images is computed as:

$$N_h = (\text{oldHeight} + 1) / 4 \quad (4a)$$

$$N_w = (\text{oldWidth} + 1) / 4 \quad (4b)$$

$$N_s = \text{NewWidth} * \text{NewHeight} \quad (4c)$$

Where N_h is new height, N_w is new width and N_s is new size.

ii. Convert the source image from RGB to HSV

Resulting image after downsample operations is converted from RGB to HSV color space such explained before.

iii. Convert HSV image to binary image

HSV image is converted to a binary image. The searching for pixel values which have an exact matching to the Mean HSV value of the detected faces in that previously stored is the tiring and impractical process to find a face in the frame. Therefore, a threshold value is used in the matched pixels searching operation. Minimum and maximum value will be defined for each three mean components (HSV); instead of the comparison with a single value for each components of the Mean HSV, the comparison will be with a range of values between the maximum value and the minimum value.

The pixel value in the binary image will be **1** if the value of the corresponding pixel in the image lies between the maximum and minimum value, while it is **0** if it is not within this range.

$$\text{HSVMax} = [\text{MHSV.H} + 25, \text{MHSV.S} + 50, \text{MHSV.V} + 50] \quad (5a)$$

$$\text{HSVMin} = [\text{MHSV.H} - 25, \text{MHSV.S} - 50, \text{MHSV.V} - 50] \quad (5b)$$

$$\text{Binaryimage}(i,j) = \begin{cases} 1 & \text{if HSVMin} \leq \text{HSVimage}(i,j) \leq \text{HSVMax} \\ 0 & \text{Otherwise} \end{cases} \quad (5c)$$

iv. Dilation: Dilates the binary image by using a 3×3 rectangular structuring element to fill holes of a size equal to or smaller than the structuring element.

v. Contours: Finding the contours in a binary image using the border following algorithm.

3.6 Blurring faces

It is a process of masking the faces that appear on the screen, in order to hide them when seeing them. In this research, Gaussian filter has been used to blurring the faces. The process of faces blurring is used in two different step, these are:

i. Blurring the currently detected faces

Each face currently detected by the Viola Jones algorithm will be directly blurred by using Gaussian filter.

ii. Blurring selected areas to be faces

Implement the blurring process by using Gaussian filter for each area that has been identified as a face.

4. Experiments and Environment

Experiments depending on the two types of parameters that determine the type of test. These parameters are:

i. Background:

- **Dynamic Background (Uncontrolled):** The background does not comply with any condition and may contain movement or color which affect the results.

ii. Number of faces.

- **Single face:** The test is performed on only one face in the scene. The results are recorded based on the system's ability to detect, track and blur this face.

- **Multiple faces:** The results will depend on the ability of the proposed system to detect several faces at once and then track and blur them.

5. Result of the system

The execution time and the accuracy of face blurring in uncontrolled background and single face are **68ms** and **82.8%** respectively, where 100fps is the average number of frames for the frontal and the profile faces where detected at execution time. There are some error happened in treatment operation such as in image number 7, which detect the face but blur was failed and tracing step covered a part of the face. In image number 9, the mean HSV values that used is H=25, S=50, V=50 which it was more sensitive to the colours and blur all the red background even the face, but by try and error the mean HSV that gave the best result is H=10, S=20, V=20 in image number 8. Figure-3 shows face blurring in dynamic (uncontrolled) background and single face:



Figure 3-number of face blurring frames in uncontrolled background and single face.

Table 2-the accuracy and time spent for a number of face blurring frame in uncontrolled background and a single face

No.	Frames (fps)	Blurring accuracy %	Total time (ms)
1	10	90	69
2	100	79	67
3	200	87	68

The second result of execution time and the accuracy of face blurring in uncontrolled background and multiple faces is **69.7ms** and **73.85%** respectively where 100fps is the average number of frames for the frontal and the profile faces where detected at execution time. Where, there are some error happen in image number 2, because of the fast motion where the detection and the tracking failed to blur the face shown in Figure-4:

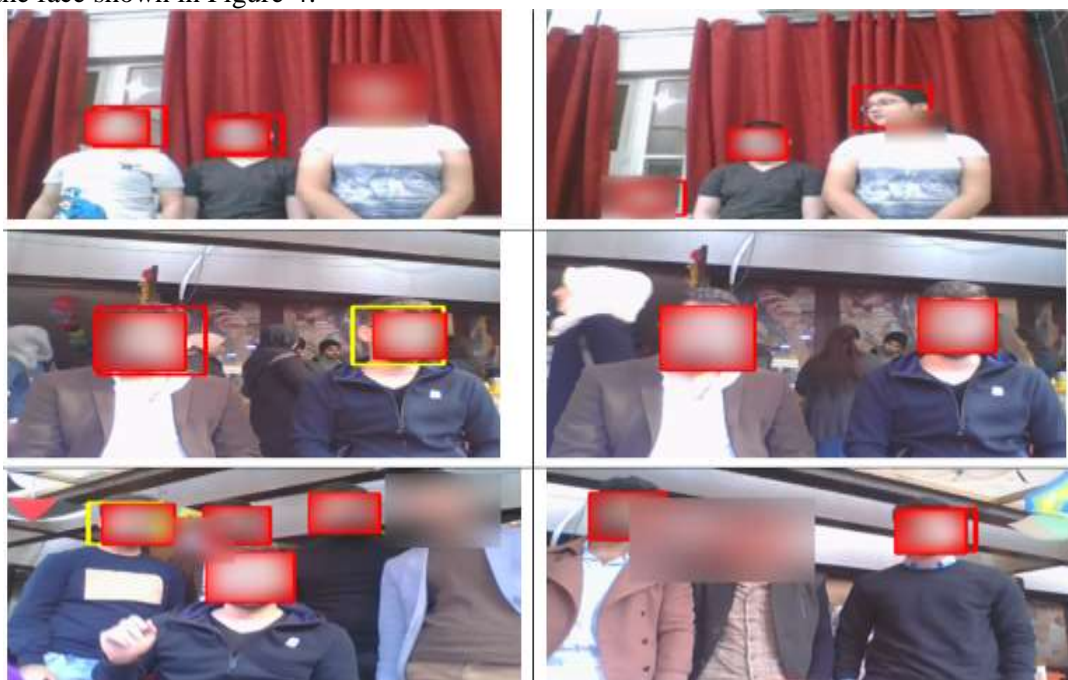


Figure 4-face blurring in uncontrolled background and multiple faces.

Table 3-the accuracy and time spent for a number of face blurring frames in uncontrolled background and multiple faces

No.	Frames (fps)	Blurring accuracy %	Total time (ms)
1	10	83	71
2	100	70	70
3	200	79	76

6. CONCLUSION

The proposed system presented an online face detection, tracking and blurring operation is done, where 100fps is used with uncontrolled environment. In a single face, the average execution time around 68ms with an accuracy of 85.3% on frontal and profile face detection, tracking and blurring and the average execution time for multiple faces is 71.6ms with an average of accuracy 77.3%, from this it observed that the accuracy and time are not fixed due to the effect of dynamic background on the image. The spending time increased whenever the number of faces increased. The accuracy decreased when there is a fast face motion. In color space algorithm, the mean HSV is the most effective parameter on the blurring operation, where the best selective values for HSV parameters are (H = 10, S = 20 and V = 20).

References

1. Gutta, S., Trajkovic, M., Colmenarez, A. J. and Philomin, V. **2005**. *Method and apparatus for automatic face blurring*. United States Patent Google Patents.
2. Asmita, S., Agarwal, S. and Sethy, P. **2013**. Real Time Simple Face-Tracking Algorithm based on Minimum Facial Features. *International Journal of Computer Applications*, **79**(12): 28-34.
3. Rout, R. K. **2013**. A survey on object detection and tracking algorithms. M.Sc. thesis, National Institute of Technology Rourkela.
4. Niazi M. and Jafari Sh. **2010**. Hybrid Face Detection in Color Images. *IJCSI International Journal of Computer Science*, **7**(5): 367-373.
5. Makovetsky, R. and Petrosyan, G. **2013**, Face detection and tracking with web camera. M.Sc thesis . *Fundamentals of Computer Vision* .
6. Hendi, S. H. and Mohammed, F. **2016**. Robotic System for Tracking Moving Objects Based On Their Color. *Iraqi Journal of Science*. **57**(3A): 1833-1845.
7. Yee, E. **2017**. Cascaded Facial Detection Algorithms To Improve Recognition. M.Sc. Thesis , San Jose State University (SJSU).
8. Al-Mukhtar, F. **2017**. Tracking and Blurring the face in a video file. *Al-Nahrain Journal of Science*. (1): 202-210.
9. Vijayarani, S. and SAKILA, M. A. **2015**. Template matching technique for searching words in document images. *International Journal on Cybernetics & Informatics (IJCI)*, **4**(6): 25-35.
10. BAE, J. S. and SONG, T. L. **2008**. Image tracking algorithm using template matching and PSNF-m. *International Journal of Control, Automation, and Systems*, **6**: 413-423.
11. SONKA, M., HLAVAC, V. and BOYLE, R. **2014**. *Image processing, analysis, and machine vision*, 3rd Edition, Cengage Learning.