A Visitor Counter System Using Fuzzy Measure Theory and Boosting Method

M. RAHMAT WIDYANTO, FERDI FERANDI, MARTIN LEONARD TANGEL
Faculty of Computer Science
University of Indonesia
Depok Campus, Depok 16424, West Java
INDONESIA
widyanto@cs.ui.ac.id, ferdi.ferandi@ui.edu, martin.tangel@gmail.com

Abstract: - This paper proposes a visitor counter system which is capable of counting single & multiple object visitors using fuzzy measure theory for tracking and Boosting method for classification. Besides fuzzy measure theory, the paper also uses Euclidean distance to track a visitor based on his movement between each frame, while fuzzy measure theory tracks a visitor based on trust degree. Both system performances are compared for their visitor tracking accuracy and their computational speed. Experimental results show that Euclidean distance and Fuzzy measure have similar accuracy for tracking visitor. However, Euclidean distance is faster than those of fuzzy measure theory in the computational speed. The proposed visitor counter system can be further developed for real-time visitor counting in shopping mall, station, and other places.

Keywords: - boosting classifier, object detection, fuzzy measure, euclidean distance, tracking, counting

1 Introduction

In a public place such as shopping malls and cinemas, data on the number of visitor is frequently needed for marketing research or statistic purposes. Usually the counting process is done manually by the officers who guard the entrance [1]. If this process is done for a long period of time, it will be prone to human errors.

To overcome this problem, a system which is able to count automatically should be developed. The system will work in real time and is integrated to the CCTV camera placed at strategic places. This system must be able to detect visitor and track their appearance on the video, and finally count the number of the visitors. Tracking process is essentially related to the visitor counting process.

Before the system tracks and counts the visitors, it should detect the visitor objects in the first stage. In earlier research, one of the methods that was often used for object detection is wavelet template with AdaBoost training. Viola and Jones [2] use this method and propose a rapid object detection scheme based on multi-stage simple feature classifier. Wavelet template method is a method to extract image features. The advantage of using this method is its fast computation time, but the detection accuracy is not high. Boosting method, which is also known as AdaBoost, is a powerful method to train a classifier which can be used to decide if an object is classified as a class or not. A Wavelet template method combined with AdaBoost training method will result an integrated method which has both good detection accuracy and fast computation time.

The visitor counter system uses the wavelet template and Boosting training method for visitor object detection. For tracking visitor, the system will implement 2 different methods; they are Euclidean distance and Fuzzy measure. These 2 methods will be compared for their detection performance and computation speed.

Section 2 will discuss the method which was implemented in the visitor tracking module, Fuzzy measure and Euclidian distance. Boosting as the training method will be explained in section 3. Visitor counter architecture will be presented in section 4. Experimental result of the visitor counter system is shown in section 5.

2 Tracking Method

This section will explain the fundamental theory of Fuzzy measure and Euclidean distance. Both of these methods are used as the methods for tracking objects in this research.

2.1 Fuzzy Measure Theory

Fuzzy measure theory is used in image understanding to combine information from several sources [3]. In this approach, the information sources are given grades of compatibility, and their evidence is weighted and combined accordingly. Fuzzy measure obeys the following properties. Let X be any set, let $P(x)$ be the power set of X, and let $g : P(x) \rightarrow [0,1]$. This g is called a fuzzy measure if $\forall A, B, A_i \in P(X)$

$$g(\phi) = 0\ ;\ g(X) = 1 \tag{1}$$

$$g(\phi) \ge g(A)\text{, if } B \supset A \tag{2}$$

If $\{A_i\}_{i=1}^{\infty}$ is monotonic,

$$\text{then } \lim_{i\to\infty}\{g(A_i)\} = g\left(\bigcup_{i=1}^{\infty} A_i\right) \tag{3}$$

When g satisfies (4), it is called a λ-fuzzy measure, and written as g_λ instead of g

$$g_\lambda(A \cup B) = g_\lambda(A) + g_\lambda(B) + \lambda g_\lambda(A) g_\lambda(B) \tag{4}$$

In practice, a good value for λ in (4) must be determined, given the value of the measure for the singletons in X, i.e., a value which is useful for this dynamic image understanding application. Let $X = \{x_1, x_2, ..., x_n\}$, and let $g^i = g_\lambda(\{x_i\})$. The values $\{g^i : i = 1,2,...,n\}$ are called the fuzzy densities associated with X, and are interpreted as the importance of the individual (singleton) information sources [4]. Now, suppose $A \subset X$, say, $A = \{x_{i_1}, ..., x_{i_m}\}$. A is viewed as a set of information sources, and the g_λ measure of A $g_\lambda(A)$ is regarded as the importance of that subset of sources for answering some questions. The measure of A is [5]:

$$g_\lambda(A) = \sum_{j=1}^{m} g^{i_j} + \lambda \sum_{j=1}^{m-1} \sum_{k=j+1}^{m} g^{i_j} g^{i_k} + \ldots + \lambda^{m-1} g^{i_1} g^{i_2} \ldots g^{i_m} \tag{5}$$

For $\lambda \neq 0$ and $A = X$, this equation can be rewritten as:

$$g_\lambda(X) = \frac{1}{\lambda}\left[\prod_{i=1}^{n}\left(1 + \lambda g^i\right) - 1\right] \tag{6}$$

The value of λ can be calculated from this expression aftermath, given the definition of fuzzy measure, $g_\lambda(X) = 1$

$$1 + \lambda = \prod_{i=1}^{n}\left(1 + \lambda g^i\right) \tag{7}$$

In [6], it is shown that there is a unique solution for this expression for $\lambda > -1$ and $\lambda \neq 0$.

When g is a g_λ-measure, the values of $g_\lambda(A_i)$ can be determined recursively as [7]

$$g_\lambda(A_1) = g_\lambda(\{x_1\}) = g^1 \tag{8}$$

$$g_\lambda(A_1) = g^1 + g_\lambda(A_{i-1}) + \lambda g^i g_\lambda(A_{i-1}) \quad 1 < i \le n \text{ and } A_i = x_1, \ldots x_i \tag{9}$$

The result g_λ is regarded as the belief assignable to an object regarding its membership in a class given the densities from each of the sources.

2.2 Euclidean Distance

Euclidean distance is the shortest distance between two points, which is measured by drawing a straight line to connect those two points. In N-dimensional space, R^n the distance between x and y point can be written as follows [8]:

$$D = |x - y| = \sqrt{\sum_{i=1}^{n} |x_i - y_i|^2} \tag{10}$$

Because the developed system will work in two dimension, the Euclidean distance between p (x_1,y_1) and q (x_2,y_2) can be computed as:

$$D(p,q) = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \tag{11}$$

In the visitor counter system, the Euclidean distance will be measured from the center point of detected visitor on one frame to the center point in the subsequent frame.

3 Boosting

Boosting [9] is a powerful technique for combining multiple base classifiers to produce a form of committee whose performance can be significantly better than that of any the base classifier. The most widely used form of boosting algorithm called

AdaBoost, shot for 'adaptive boosting', developed by Freund and Scaphire [10]. Boosting can give good results even if the base classifiers have a poor performance, and hence sometimes the base classifers are known as weak learners [11].

The principal of boosting is that the base classifiers are trained in sequence, and each base classifier is trained using a weighted form of the data set where the weighting coefficient associated with each data point depends on the performance of the previous classifiers.

In particular, points that are missclasiffied by one of the base classifiers are given greater weight when used to train the next classifier in the sequence. Once all the classifiers have been trained, their predicitons are then combined through a weighted majority voting scheme, as illustrated schematically in Figure 1.

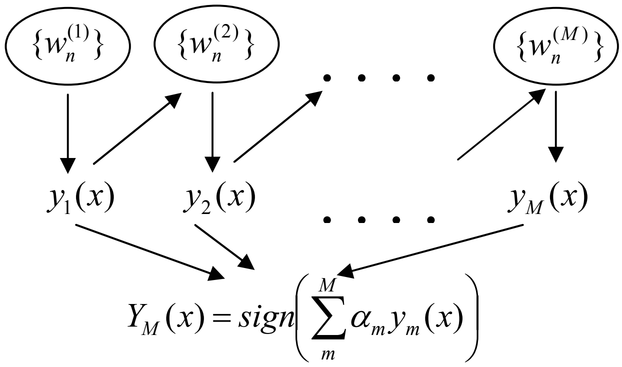

Fig 1. Schematic illustration of the boosting framework

Consider a two-class classification problem, in which the training data comprises input vectors x1,....., xN along with corresponding binary target t1, t2,......, tN where tn ∈ {-1,1}. Each data point is given an associated weighting parameter wn, which is initially set to 1/N for all data points.

At each stage of the algorithm, AdaBoost trains a new classifier using a dataset in which the weighting coefficients are adjusted according to the performance of the previously trained classifier so as to give greater weight to the misclassified data points. Finally, when desired number of base classifiers has been trained, they are combined to form a committee using coefficients that give different weight to different base classifiers. The precise form of the AdaBoost algorithm is explained within 3 stages, they are input stage, initialization stage (including the updating weight of training data), and finally the output stage. The detail explanation of this algorithm is as follows:

AdaBoost algorithm

---

1. **Input:** a set of training data with labels

2. **Initialize**: the weight of training data: $w_n^{(m)} = 1/N$ for n = 1, 2, ….., N.

3. **Do For** m = 1, ….., M

(a) Fit classifier $y_m(\mathbf{x})$ to the training data by minimizing the weighted error function

$$J_m = \sum_{n=1}^{N} w_n^{(m)} I(y_m(x_n) \neq t_n) \qquad (1)$$

where $I(y_m(x_n) \neq t_n)$ is the indicator function and equals 1 when $y_m(x_n) \neq t_n$ and 0 otherwise.

(b) Calculate the training error of y_m:

$$\varepsilon_m = \frac{\sum_{n=1}^{N} w_n^{(m)} I(y_m(x_n) \neq t_n)}{\sum_{n=1}^{N} w_n^{(m)}} \qquad (2)$$

(c) Set weight of base classifier y_m:

$$\alpha_m = \ln\left\{\frac{1-\varepsilon_m}{\varepsilon_m}\right\} \qquad (3)$$

(d) Update the weight of training data:

$$w_n^{m+1} = w_n^m \exp\{\alpha_m I(y_m(x_n) \neq t_n)\} \qquad (4)$$

3. **Output**:

$$Y_M(x) = sign\left(\sum_{m=1}^{M} \alpha_m y_m(x)\right) \qquad (5)$$

---

4 System Architecture

The visitor counter system was built upon the Open Source Computer Vision library for face detection [12]. There are 2 types of visitor counter system implemented. One for counting human visitors, and another for tracking multiple visitor object, humans, motorcycles and cars. For the human visitor counter, it was assumed that the face is the most important part of a visitor, so it will be the part that will be tracked and counted by the system. While for the multiple object counter, it will track the shape of humans, motorcycles and cars. Before the system is capable to count visitor, firstly, it has to be trained to detect if there is a specific object in the video frames. Because of that, the system architecture will be divided into two parts, they are training architecture and visitor counter architecture.

4.1 Training Architecture

In the training architecture, there are 2 types of samples which will be used for training samples; they are positive and negative samples. The positive samples for human visitor counter consist of images which have a face in it with 20 x 20 pixels resolution. The positive samples for multiple visitor counters consist of side view images of humans, motorcycles and cars. The dimensions for human object is 20 x 50 pixels, 38 x 38 pixels for motorcycles, and 50 x 25 pixels for cars. The image training dimensions were chosen based on the similarity to the dimension of detected object in the experiment video later, which also depends on the camera placement. The negative samples for both systems consist of background images that contain no positive samples.

Both sample image feature will be extracted. The features used are based on Haar wavelets, which called Haar feature. The two-dimensional Haar decomposition of a square image with n^2 pixels consist of n^2 wavelet coefficients, each of which corresponds to a distinct Haar wavelet [13]. The first wavelet is the mean pixel intensity value of the whole image. The rest of the wavelets are computed as the difference in the mean intensity values horizontally, vertically, or diagonally adjacent squares. These Haar features will become the inputs for the training process.

Each object will be trained separately. The training process will be using Boosting method. The Result of the training process is a cascaded classifier. An image will be classified as detected object if it passes all of the layers in the cascaded classifier. Using cascaded classifier means that each base classifier doesn't need to have a very accurate detection, but the combination of all basic classifier will be highly accurate. For example, if one base classifier has 0.999 hit rate and 0.5 false alarm, 20 cascaded classifier will have $0{,}999^{20} \approx 0{,}98$ hit rate dan $0{,}5^{20} \approx 10^{-6}$ false alarm.

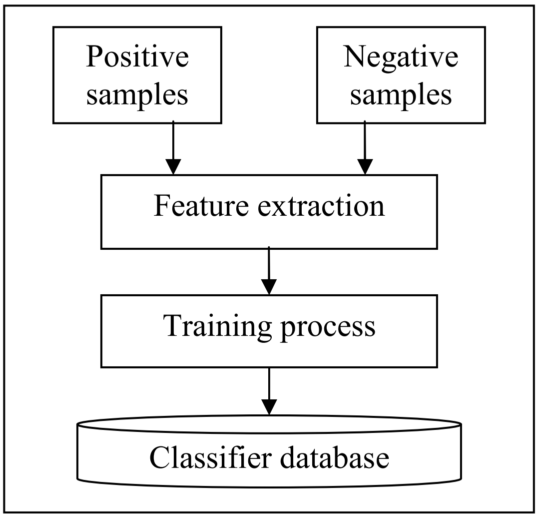

Fig 2. The schema of the training process

For human visitor counter system, assuming that a face is the most important part while detecting human visitor, the positive samples will be front view of human faces. For multiple visitor object counter, the positive objects will be side view images of humans, motorcycles, and cars. The negative samples are background images which doesn't contain the positive samples.

4.2 Visitor Counter Architecture

There are 3 sub architectures in the visitor counter architecture; they are detection, tracking and counter.

4.2.1 Detection Architecture

The input file is a video file or a real-time video streamed from a security camera and also the cascaded classifier from the earlier training process.

Detection process is done by sliding a search window through the image and checking whether an image region at a certain location looks similar to a positive sample or not by filtering with the cascaded classifier from the training process. To detect an image that might contain a positive object with different size, the classifier has the ability to scale its size [12].

For multiple object counters, the search window will be checked by 3 classifiers. If the object is detected by 2 classifiers, the classifier which has the highest likeliness will be assumed as the correct object class.

The detection output for each frame is a center coordinate and radius of each detected object. The coordinate and radius will be used for the tracking process in the next frame.

4.2.2 Tracking Architecture

Tracking process tracks the movement of a detected object from one video frame to the other frame. The result of detection process earlier is a circle mark at the image region which contains a positive object, along with coordinate of the circle's centre and also radius length which show the height and width of the positive object. The tracking process will be done to the centre of the positive object.

To be able to track positive object, it was necessary to decide if the object detected in the current frame is the same object from the earlier frame. Because of that reason, information of the detected objects in each frame will be recorded. The system doesn't know the identity of each object (detection, not recognition), therefore the information recorded will be the centre coordinate of the object and its radius.

Tracking process will match an object detected in the current frame to the previous frame. The tracking process uses 2 methods, Euclidean distance and Fuzzy measure which are explained in section 2 earlier. The result performance of these 2 methods will be compared in section 5.

For tracking, an array will be used to keep track the information of each object in each frame.

Visitor	Frame 1	Frame 2	Frame 3
# 1	(x,y) = (3,4) Radius = 22	(x,y) = (4,5) Radius = 25	?
# 2	(x,y) = (100,100) Radius = 32	(x,y) = (104,107) Radius = 29	?
# 3	(x,y) = (204,103) Radius = 44	(x,y) = (209,120) Radius = 51	?

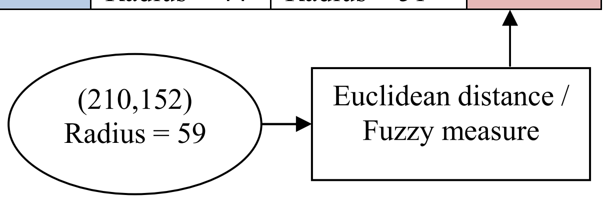

Fig 3. Illustration of Array which records visitor information.

Figure 3 illustrates the object tracking process between each frame. In the array illustration, the current frame is the third frame, and it's also recorded that there are 3 objects detected in the first and second frame. In the third frame, currently detected an object which has (210,152) coordinate and 59 pixel radius. Both of this information will be used as inputs to the Fuzzy measure or Euclidean distance process. The process will then decide, to which object the detected one actually is. A correct processing will decide that the detected object is matched with Visitor 3, because from the coordinate and radius information, it's matched best with visitor 3 in the previous frame. In the Euclidean distance process, the matching will be decided by the shortest path distance, while fuzzy measure process will decide it according to the maximum trust degree.

4.2.3 Counter Architecture

After tracking the object in each frame, visitor will be counted. The method used is different from Oliver Silda et al. [14], in this system, a visitor will be assumed entered the room if one fulfills all of these conditions:

1. Detected in 4 previous subsequent frames.
2. Undetected in the current frame.
3. Is inside the counting zone in the previous frame (last detected).

The first condition makes sure that the identified object is not a false positive, thus the object needs to be detected in 4 subsequent frames. This system only uses the last 4 subsequent frames, because if the number of frames is too many, the possibility of lost detection will be higher and will affect the performance of the object counter.

The second condition is the detected object is not detected anymore in the current frame, so the visitor is assumed entered the room if the third condition is also fulfilled. Counting zone (also known as a virtual gate [14]) in the third condition is a predetermined area and it depends on the room positioning. In reality, the counting zone usually is placed near to the door.

Figure 4a and figure 4b illustrate a counting zone. The counting zone in this image is placed at the right side, because visitor comes from the stairs in the middle and walk to the right side of the camera. If the camera was moved to another place, the counter zone will also need to be adjusted.

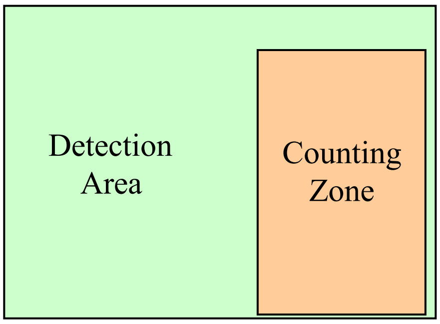

Fig 4a. An illustration of the counting zone.

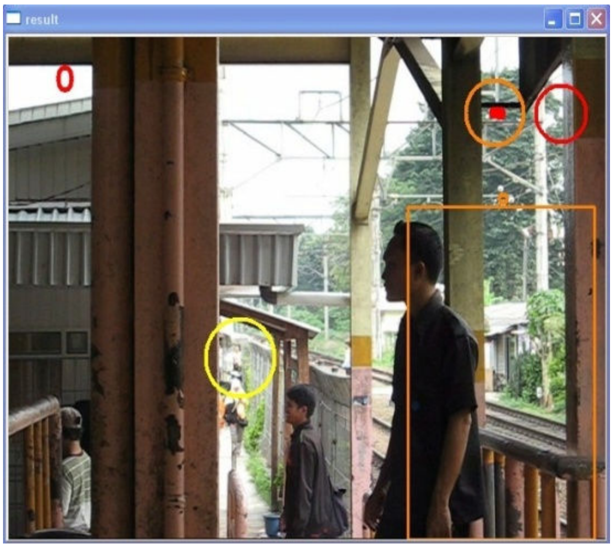

Fig 4b. The counting zone in the system.

However, the counting architecture requires a good detection capability. If a visitor is missed during detection in one frame, it might cause the visitor is not counted at all, or counted multiple times.

5 Experimental Results and Analysis

This section will discuss the data sample for testing, testing scenario, and the experimental results along with the analysis.

5.1 Testing Data

The testing used a video recorded by the author. Recording was taken in 3 different locations. The first location was inside a laboratory which consist of 13 short video, the second one was at the train station consist of 1 long video, and the third one was on the parking area consist of 1 long video. The first and second videos were used to test the human visitor counter system, and the third video was used to test the multiple object counter system.

The recording in the laboratory and at the parking area were scenes by scenario, thus the movement of the visitors are not natural. This was useful to test some movement, such as 2 people walking side by side, or 2 person walking crossing with each other. While the recording at the train station is a natural scene without scenario, thus the recorded visitors moved naturally and represent a real life condition.

5.2 Testing Scenario

In the testing scenario, the system was set to use 2 different methods to track visitor. The first method was Euclidean distance, and the second one was Fuzzy measure. These 2 methods performance and accuracy were compared.

In the testing scenario, there were 4 cases which become inputs for the testing. These cases were selected because they represent real life possibilities. Those 4 cases are:

1. A visitor walking alone.
2. Two visitors walking side by side.
3. Two visitors walking crossing each other.
4. Some visitor walking in random formation.

These 4 cases were recorded in the laboratory. The train station recording also contains all 4 cases in one recording. And the parking area recording contains case number 1, 2 and 3. Table 1 shows the recording details.

Table 1. Recording details for testing

<table>
<tr><th>Id</th><th>Visitor Numbers</th><th>Duration (mm:ss)</th><th>Size (pixel)</th><th>Case Scenario</th><th>On Scenario</th></tr>
<tr><td>1</td><td>1</td><td>00 : 12</td><td rowspan="4">320 x 240</td><td>1</td><td rowspan="4">Yes</td></tr>
<tr><td>2</td><td>2</td><td>00 : 17</td><td>2</td></tr>
<tr><td>3</td><td>2</td><td>00 : 16</td><td>3</td></tr>
<tr><td>4</td><td>3</td><td>00 : 20</td><td>4</td></tr>
<tr><td>5</td><td>21</td><td>04 : 16</td><td>640 x 480</td><td>1,2,3,4</td><td>No</td></tr>
<tr><td>6</td><td>17</td><td>01 : 22</td><td>512 x 384</td><td>1,2,3</td><td>Yes</td></tr>
</table>

5.3 Experimental Results

From the experiments, Fuzzy measure and Euclidean distance have the same accuracy, so the count results are similar. The difference is only the computation time.

For recording 1, 2, 3 and 4 which was taken in the laboratory on scenario, the system was able to count visitors correctly for each experiment cases. For recording number 5 which was not on scenario, there are 21 visitors entering the train station. In this experiment, the system was able to count 13 visitors correctly. The other 8 detected visitors were false negatives, where there was a visitor to count but the system missed it. These 8 false negatives will be explained in table 2. The system also has 3 cases of false positives, where actually there was no visitor to count, but the system detected it as a visitor and increased the counter. Description about False positives is explained in table 3. The experiment result for all visitors in recording number 5 is listed in table 4.

Table 2. False negative for train station recording.

# Visitor	Count Status	Description
1	Failed (false negative)	Visitor was not detected
2		Visitor turn back at the entrance
6		Visitor was not detected
9		Visitor was not detected
11		Visitor's height is too high, his face passed above the counting zone
13		Visitor was not detected
16		Visitor was not detected
17		Visitor's height is too high, his face passed above the counting zone

Table 3. False positive for train station recording.

Counter Number	Count Status	Description
1	Failed (false positive)	False positive detected at the counting zone, counted as visitor
7		
15		Tracking for visitor number 19 was missed in a frame at the counting zone, thus tracked and counted for the second time

Table 4. Experiment result for all visitors in train station recording.

# Visitor	Count Status	Count Number
1	False negative	0
	False positive	1
2	False negative	2
3	Success	3
4	Success	4
5	Success	5
6	False negative	5
7	Success	6
	False positive	7
8	Success	8
9	False negative	8
10	Success	9
11	False negative	9
12	Success	10
13	False negative	10
14	Success	11
15	Success	12
16	False negative	12
17	False negative	12
18	Success	13
19	Success	14
	False positive	15
20	Success	16
21	Success	17

For recording 6 for multiple object counter experiment, there were 17 visitors passing the parking area. In this experiment, the system was able to count 15 visitors correctly. There was several case of false negative case of undetected visitor and false positives case where there was no visitor but detected as a visitor, which will be explained in table 5. The experiment result for all visitors in recording 6 is listed in table 6.

Table 5. False detection for parking area recording.

# Visitor	Count Status	Description
After 9	Failed (false positive)	Human visitor turned back in the counting zone without entering the door.
11	Failed (false negative)	A Car was counted twice.
15		A Car passed by too fast, detected only in 3 frames.
After 16	Failed (false positive)	The tracking was missed in a frame in counting zone, thus it was counted as a visitor.

Table 6. Experiment result for parking area recording.

# Visitor	Count Status	Count Number
1. Motorcycle	Correct	1
2. Human	Correct	2
3. Motorcycle	Correct	3
4. Car	Correct	4
5. Motorcycle	Correct	5
6. Human	Correct	6
7. Motorcycle	Correct	7
8. Motorcycle which stopped in a middle for a while, then passed the area	Correct	8
9. Motorcycle passing by a car.	Correct	9
Human which turned back, shouldn't be counted	Mistrack in a frame and counted multiple times	12
10. Motorcycle stopped in the middle and passing by a car	Correct	13
11. Car which stop and passed by several visitor, then enter the area	Tracked correctly, but missed in a frame and detected multiple times	16
12. Motorcycle	Correct	17
13. Car	Correct	18
14. Motorcycle	Correct	19
15. Car	Not detected since it passed too fast	19
16. Human	Correct	20
Human walking to the opposite side, shouldn't be counted	The tracking missing in a few frame and it was counted multiple times	22
17. Motorcycle	Correct	23

Based on the experiment, failed countings were caused by:

1. False negative in face detection.
2. False positive in face detection.
3. Visitor face was outside the counting zone.
4. Visitor turned back at the entrance.

In the experiment, it shown that visitor counter using Fuzzy measure and Euclidean distance have same counting accuracy. If the detection capability can be increased, the counting accuracy will also be improved. This can be done by increasing the amount of training samples, eventhough this will also increase the training times. The differences in computation time for human visitor counter will be described in table 7 and 8, and the differences for multiple object counters will be described in table 9.

Table 7. Fuzzy measure and Euclidean distance computation time for human visitor counter (ms).

Frame Size	Parameter	Fuzzy measure	Euclidean distance	Diff.
320 X 240	Average	0.00345	0.00049	0.00295
	Minimum	0.00230	0.00014	0.00213
	Maximum	0.05355	0.00163	0.05192
640 X 480	Average	0.00293	0.00032	0.00261
	Minimum	0.00224	0.00012	0.00206
	Maximum	0.02157	0.00016	0.02141

Table 8. Fuzzy measure and Euclidean distance computation time between frames for human visitor counter (ms).

Frame Size	Parameter	Fuzzy measure	Euclidean distance	Diff.
320 X 240	Average	75.11624	73.43745	1.678791
	Minimum	70.7146	69.069	1.6456
	Maximum	79.7919	79.4814	0.3105
640 X 480	Average	334.9573	333.5488	1.408559
	Minimum	316.721	275.339	41.382
	Maximum	435.492	432.64	2.852

Table 9. Fuzzy measure and Euclidean distance computation time for multiple object counter (ms).

Object (pixels)	Parameter	Fuzzy measure	Euclidean Distance
Human (20x50)	Average	0.02334	0.01607
	Minimum	0.02218	0.01485
	Maximum	0.02641	0.01797
Motorcycle (38x38)	Average	0.04667	0.04527
	Minimum	0.04362	0.04157
	Maximum	0.04729	0.04729
Car (50x25)	Average	0.03416	0.03134
	Minimum	0.03081	0.03010
	Maximum	0.03754	0.03548
Combination	Average	0.09758	0.09388
	Minimum	0.09386	0.09082
	Maximum	0.10489	0.09635

In the computation time for both single object and multiple object counters, Euclidean distance is faster because Fuzzy measure also uses Euclidean distance as a Fuzzy measure input. In tracking speed per video frame, Euclidean distance is also faster than Fuzzy measure. Eventhough Fuzzy measure is slower in computation time compared to Euclidean distance, theoretically it also has advantages:

1. Fuzzy measure also calculate face radius as an input.
   Fuzzy measure track visitor's face based on trust value which is calculated by using Euclidean distance and face radius difference as parameters. While Euclidean distance only uses distance to track faces, fuzzy measure can anticipate the problem of an abnormal change in face radius, thus it will not be counted as a same visitor, while Euclidean distance will assume it is the same visitor. This advantage doesn't appear in the experiment, because this condition rarely happens.
2. Fuzzy measure is easier to adapt.
   As explained before, if the camera position is moved, the system will need to be adjusted again. Fuzzy measure will be easier to adapt because it use intuitive membership function which make it easier to adjust the rule.

6 Conclusions

Based on experimental result and analysis, the summary of visitor counter system and the method used are as follows:

1. Either Fuzzy measure or Euclidean distance have same tracking performance and counter accuracy in the experiment.
2. For calculation speed, Euclidean distance is faster than Fuzzy measure. This is because Fuzzy measure needs to count Euclidean distance first. The calculation time difference is around 0.003 ms on average.
3. Visitor counter system works correctly. Incorrect visitor counting that happens in the experiment was caused by faulty object detection. In the experiment, it is shown that if a visitor detected correctly, the tracking and counting process will also work correctly.

The proposed visitor counter system can be further developed for real-time visitor counting in shopping mall, station, and other places.

References:

[1] Sriwijaya Post, "Mal Raup Miliaran Rupiah," Thursday, October 4th 2007, pp. 15.

[2] Paul Viola, Michael Jones, Rapid Object Detection Using a Boosted Cascade of Simple Features, *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, ISSN: 1063-6919, Vol. 1, December 2001, pp. 511-518.

[3] Maria Susan Anggraeiny, *Representasi Kueri Spasial Warna Dengan Logika Fuzzy pada Sistem Perolehan Citra*, Tesis S2, Fakultas Ilmu Komputer, Universitas Indonesia, 2007.

[4] Li Wang, Hai Hong Wang, and Xiao Peng Ji. Multi Stage Moving Object Recognition Based on Fuzzy Integral, *College of Information Science and Engineering*, Ocean University of China, 2007.

[5] Zhi-Qiang Liu, Leonard T. Bruton, James C. Bezdek, James M. Keller, Sandy Dance, Norman R. Bartley, and Cishen Zhang. Dynamic Image Sequence Analysis Using Fuzzy Measures, *IEEE Transactions on Systems, Man, and Cybernetics*, 2001, pp. 557 – 572.

[6] Janusz Kacprzyk, Anna Wilbik, and Slawomir Zadrozny, Analysis of Time Series via Their Linguistic Summarization: The Use of the Sugeno Integral, *IEEE Seventh International Conference on Intelligent Systems Design and Applications*, 2007, pp. 262 – 267.

[7] H. Tahani and J.M. Keller, *Information Fusion in Computer Vision using The Fuzzy Integral, in Fuzzy Measure Theory*. New York: Plenum, 1992, pp. 319 – 341.

[8] Nova Eka Diana, *Sistem Temu Kembali Citra Wajah Berbasis Eigenface*, Tugas Akhir S1, Fakultas Ilmu Komputer, Universitas Indonesia, 2007.

[9] Christopher M. Bishop, *Pattern Recognition and Machine Learning*, Springer, 2006.

[10] Yoav Freund, Robert E. Schapire, A decision-theoretic generalization of on-line learning and an application to boosting, *Journal of Computer and System Sciences*, Vol.55, No. 1, 1997, pp 119-39.

[11] Kai O. Arras, Óscar Martínez Mozos, Wolfram Burgard, Using Boosted Features for the

Detection of People in 2D Range Data, *Proc. of the IEEE International Conference on Robotics and Automation (ICRA)*. Rome, Italy, 2007.

[12] Gary Bradski, Adrian Kaehler, and Vadim Pisarevsky, Learning-based computer vision with Intel's open source computer vision library, *Intel Technology Journal*, vol. 9, 2005, pp. 119–130.

[13] Jacob Whitehill, Christian W. Omlin, Haar features for FACS AU recognition, *In: FGR 2006 Proceedings of the 7th International Conference on Automatic Face and Gesture Recognition*, 2006, pp. 97–101. IEEE Computer Society Press, Washington, DC, USA.

[14] Oliver Sidla, Yuriy Lypetskyy, Norbert Brändle, Stefan Seer, Pedestrian Detection and Tracking for Counting Applications in Crowded Situation, *Proceedings of the IEEE International Conference on Video and Signal Based Surveillance*, 2006.