Estimation of Crowd Density in Public Areas Based on Neural Network

Gyujin Kim[1], Taeki An[2] and Moonhyun Kim[1]
[1] College of Information and Communication Engineering, Sungkyunkwan University
440-746, 2066 Seobu-ro, Jangan-gu, Suwon-si, Gyeonggi-do, Republic of Korea
[2] Metropolitan Tranportation Research Center, Korea Railway Research Institute
437-757, 176 Cheoldo bangmulgwan-ro, Uiwang-si, Gyeonggi-do, Republic of Korea
[e-mail: heaven_84@naver.com, mhkim@ece.skku.ac.kr, tkahn@krri.re.kr]
*Corresponding author: Moonhyun Kim

Received December 7, 2011; revised July 27, 2012; accepted August 23, 2012; published September 26, 2012

Abstract

There are nowadays strong demands for intelligent surveillance systems, which can infer or understand more complex behavior. The application of crowd density estimation methods could lead to a better understanding of crowd behavior, improved design of the built environment, and increased pedestrian safety.
In this paper, we propose a new crowd density estimation method, which aims at estimating not only a moving crowd, but also a stationary crowd, using images captured from surveillance cameras situated in various public locations. The crowd density of the moving people is measured, based on the moving area during a specified time period. The moving area is defined as the area where the magnitude of the accumulated optical flow exceeds a predefined threshold. In contrast, the stationary crowd density is estimated from the coarseness of textures, under the assumption that each person can be regarded as a textural unit. A multilayer neural network is designed, to classify crowd density levels into 5 classes. Finally, the proposed method is experimented with PETS 2009 and the platform of Gangnam subway station image sequences.

Keywords: Crowd density, neural network, optical flow, contrast

This research was supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education, Science and Technology(2012R1A1A2008998).
http://dx.doi.org/10.3837/tiis.2012.09.011

1. Introduction

Population growth, along with the urbanization, has caused more problems in many public areas, such as subway or airport terminals, hospitals, etc. When persons gather closely together, the possibility that accidents will be caused by the crowding is very high. So, many surveillance systems have been installed in public areas, but not all of those can be monitored in real-time, because the operators that observe the monitors are very few, compared with the number of monitors. For example, an observer can miss crucial accidents or only detect them after considerable delays. Thus, intelligent systems for preventing accidents are needed, such as Intelligent Surveillance Systems.

Many researchers in the field of computer vision have studied 'People Counting' and 'Crowd Density Estimation' methods for several years. Crowd density can be measured from the number of pedestrians in a specific place. Counting the number of people through image processing requires a sophisticated combination of various techniques, such as object detection and pattern matching, hence results in heavy computational costs. These studies show that the greater the number of foreground pixels, the higher the crowd density is. A background image is used to classify image pixels as either pedestrians or background [1]. A functional relationship between the number of classified pedestrian pixels, and number of people, is calculated manually for the measurement of crowd density. Another example is proposed by Ma et al. [2], using background removal. The relationship between the number of foreground pixels and number of persons is obtained by applying a geometric correction. However, their works are only true when there are not serious occlusions between persons, with an ideal setting (i.e. proper height view, orthographic projection), perfect segmentation, and further, with the assumption of an equal size of persons.

Various features of crowds are extracted from a crowd scene to estimate the crowd density. The important features are edge, texture, optical flow, etc. These features can be extracted using computer vision techniques. Background subtraction and edge detection are used by Kong et al. [3][4]. They introduce a way of using the extracted edge orientation and blob size histograms as features. Obviously, more clues may indicate a more accurate solution for the analysis of crowd scenes. The relationship between the number of pedestrians and the feature histograms can be learned from training data.

Zhan et al. [5] present a technique that is based on differences of texture patterns of the crowd images. It assumes that images of low-density crowds tend to present coarse texture, while images of dense crowds tend to present fine textures. According to this technique, they propose a method that estimates the crowd density from the texture of the image, using Minkowski fractal dimensions [6]. Furthermore, the estimation of crowd density using texture classification is carried out by means of self-organizing maps [7]. While previous works successfully detect moving crowds, for estimating crowd density in public areas, they can hardly detect stationary crowds. Therefore, a new technique that can simultaneously detect moving crowds and stationary crowd is needed.

In this paper, we propose a new crowd density estimation method, which aims at estimating not only the crowd density of moving crowds, but also the crowd density of stationary crowds, using images obtained from surveillance cameras in various public areas. We devised two efficient features, moving area and contrast, to estimate the crowd density from the acquired video scenes. The moving crowds are estimated from the moving area, where the magnitude of the accumulated optical flow exceeds a predefined threshold. The stationary crowds are

estimated from the coarseness of textures, under the assumption that each person is regarded as a textural unit. The developed features do not require segmentation of people from background, which causes the interference problem of shadow, especially for outdoor image sequences. A 2-layer neural network is designed to classify crowd density into 5 classes. Finally the proposed method is experimented with PETS 2009 and Gangnam subway station image sequences.

The rest of this paper is organized as follows: In section 2, we explain an extraction process of moving area feature and contrast feature. For an indoor image sequence, a spatial transformation process is introduced to normalize person size, independent of his or her distance from a camera. In section 3, the designed neural network for estimation of crowd density is described, together with a training algorithm. In section 4, we show the experimental results, based on the proposed method. The neural network is trained using training sets, which are produced by labeling each frame with the number of persons in the image. By quantizing the estimated crowd density, the output of the neural network, the crowd level, is classified into 5 classes. The performance is measured as the classification accuracy of test sets. Finally in section 5, a discussion of the advantages of the proposed method and further research directions are presented.

2. Crowd Feature Extraction

In the proposed algorithm, we extract two features, moving area and contrast, from the crowd image sequences. A moving area, which is a set of pixels involved in a motion within a certain time period, is identified by using accumulated optical flow vectors. Since the optical flow is the instantaneous velocity of a pixel, we need an intelligent algorithm for processing computed optical flow, to detect pixels involved in personal motion. In order to measure the coarseness of crowd textures, we propose a gray level dependence matrix. The measured contrast is normalized to be independent of the absolute contrast of the captured image sequences.

The block diagram of our proposed neural based crowd density estimation method is shown as **Fig. 1**. For an indoor image, we introduce an additional process, extraction of ROI (Region of Interest) and spatial transformation. ROI is defined as a region where passengers pass through. Extraction of a ROI results in reducing the noise effect included in the non-ROI region, thus improving the crowd density estimation accuracy. Since the size of a person is different according to their distance from the camera, spatial transformation is applied to normalize the person size. The next step consists of an extraction process, not only from a moving crowd but also from a stationary crowd. The optical flow is calculated as a pixel wise movement vector, and the optical flow is accumulated for several frames. The region of moving crowd is identified as a set of pixels whose magnitude of accumulated optical flow exceeds a threshold.

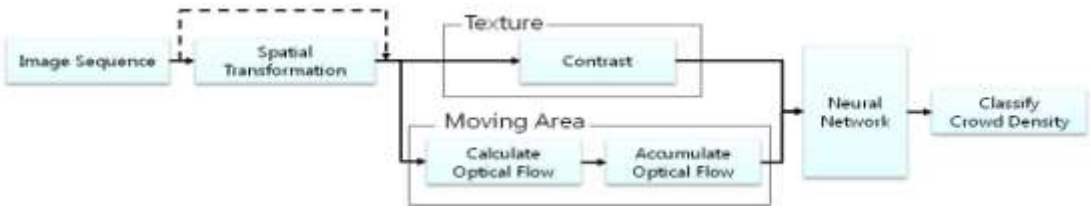


Fig. 1. The block diagram of proposed method for crowd density estimation using feature extraction and neural network classification

The moving area is measured as the ratio of pixels in the moving region to the total pixels. The feature for representing stationary crowd density is defined from texture measure, derived from a gray level dependence matrix. The feature of moving area and the feature of texture is applied to a multilayer neural network. The neural network produces an estimated crowd density as a real number from 0 to 1, where 1 denotes the most crowded situation.

2.1 Spatial Transformation

We categorized public areas into indoor areas and outdoor areas, for estimation of crowd density. We chose a platform image sequence of a subway station as an indoor environment [8]. As an outdoor environment, we chose image sequence of PETS 2009, which includes shadows resulting from the interference of objects with illumination light [9].

Fig. 2 shows images of PETS 2009 and Gangnam subway station platform. While **Fig. 2-(a)** shows that people scatter over the area of the image, **Fig. 2-(b)** shows that people do not appear beyond the yellow safety line. To prevent safety accidents, PSDs (Platform Screen Doors) have been installed beyond the safety line. Therefore, the PSD area should not be used to measure crowd density, as the right side of the yellow safety line is enough to measure the crowd density. Thus the ROI is set up as the right side of the yellow safety line.

(a) (b)

Fig. 2. (a) PETS 2009 (b) Gangnam subway station

The ROI is defined for the subway station image as in **Fig. 3**, where N and M are the number of pixels in a row and a column of an image, respectively. The slanted line is generated from the yellow safety line of platform. The left part of the line is the region that the train passes through, and is shielded by PSD. This region is removed for further analysis, since it is full of noise, due to the reflectance property of plastics used in PSDs.

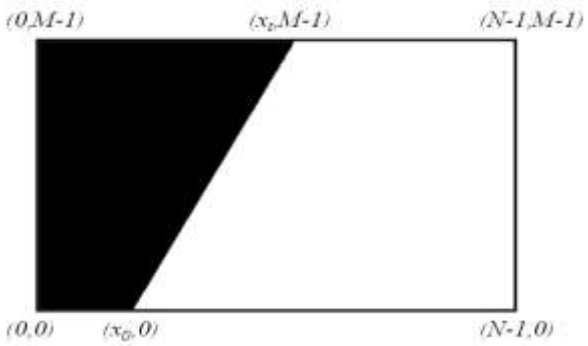


Fig. 3. Region of Interest

Since a motion vector is proportional to the person's size, it is hard to detect motion for small people in the far distance. Thus, the extracted ROI image, shown in **Fig. 4-(b)**, needs to be spatially transformed [10].

The image $f(x, y)$ is transformed to an image $g(x', y')$. Transformation is expressed as $x' = r(x, y), y' = s(x, y)$. The transformation is performed by the use of the following 4 sets of tie points. In each set, the point to the left of the arrow is a coordinate of the input point, and the point to the right of the arrow is a coordinate of a mapped point in the transformed image.

$$\begin{aligned}&(x_0, 0) \rightarrow (x_0, 0),\ (N-1, 0) \rightarrow (N-1, 0),\\ &(x_t, M-1) \rightarrow (x_0, M-1),\ (N-1, M-1) \rightarrow (N-1, M-1)\end{aligned} \tag{1}$$

Then, the transformation is modeled by the following pair of bilinear equations.

$$r(x, y) = (N-1) - \frac{\{(N-1)-x\}\{(N-1)-x_0\}(M-1)}{(M-1)\{(N-1)-x_0\} - y(x_t - x_0)},\ s(x, y) = y \tag{2}$$

(a) (b) (c)

Fig. 4. (a) Original image (b) Result of image after setting ROI (c) Spatially transformed image

In order to equalize pedestrian sizes regardless of the distance from the lens and angle of view, we applied a geometric transformation method. The transformed image is used to compute optical flow vector so that the magnitude of the optical flow vector is dependent little on the distance from the lens.

Fig. 4-(c) shows a spatially transformed image of Gangnam subway station platform. The yellow safety line is geometrically transformed to a vertical line. The size of a person becomes less dependent on the distance of the person from the camera location.

2.2 Moving Area

From the spatially transformed image, the optical flow, the velocity of each pixel is computed in order to identify regions of moving passengers. While the global method of the Horn and Schunck algorithm produces a high density of flow vectors, using a motion smoothness assumption, it has a drawback of noise sensitivity [11][12][13]. In contrast, the local method of the Lucas-Kanade algorithm is less sensitive to image noise, since it solves optical flow constraint equations for neighbor pixels, using a least square criterion. However, this method does not provide flow information for the uniform grey level regions, as computation is performed based on local region.

In this paper we need to estimate flow vector distribution for moving passengers, which requires a flow vector not only for the boundary of a person, but also for the interior region of

a person. Thus we used the so-called combined local-global (CLG) method for optical flow computation [14]. It is known that this method combines the advantages of the global Horn and Schunck approach, i.e. dense flow fields and the local Lucas-Kanade method i.e. high noise robustness.

Let $f(x,y,t)$ be an image sequence, where (x,y) denotes the coordinate of a pixel in the image plane, and t is the time when the image is captured. The CLG method computes the optical flow field $(u(x,y),v(x,y))^T$ at time t, by minimizing the following function.

$$E(u,v)=\int\int\left(\omega^T J_\rho(\nabla_3 f)\omega+\lambda(|\nabla u|^2+|\nabla v|^2)\right)dxdy \tag{3}$$

where the vector field $w(x,y)=(u(x,y),v(x,y),1)^T$ describes the optical flow field, and $u(x,y)$ denotes the x directional velocity i.e. $\partial x/\partial t$, and $v(x,y)$ denotes the y directional velocity i.e. $\partial y/\partial t$ at a pixel (x,y). The second term in equation (3) denotes the smoothness constraint of optical flow. ∇u denotes spatial gradient of x directional velocity $(u_x,u_y)^T=(\partial u/\partial x,\partial u/\partial y)^T$, while ∇v denotes the spatial gradient of y directional velocity $(v_x,v_y)^T=(\partial v/\partial x,\partial v/\partial y)^T$. The matrix $\nabla_3 f$ denotes the spatiotemporal gradient of grey levels i.e. $(f_x,f_y,f_t)^T=(\partial f/\partial x,\partial f/\partial y,\partial f/\partial t)^T$. The matrix $J_\rho(\nabla_3 f)$ is the 'motion tensor' or 'structure tensor' given by equation (4). The weight λ denotes the relative importance of the smoothness assumption.

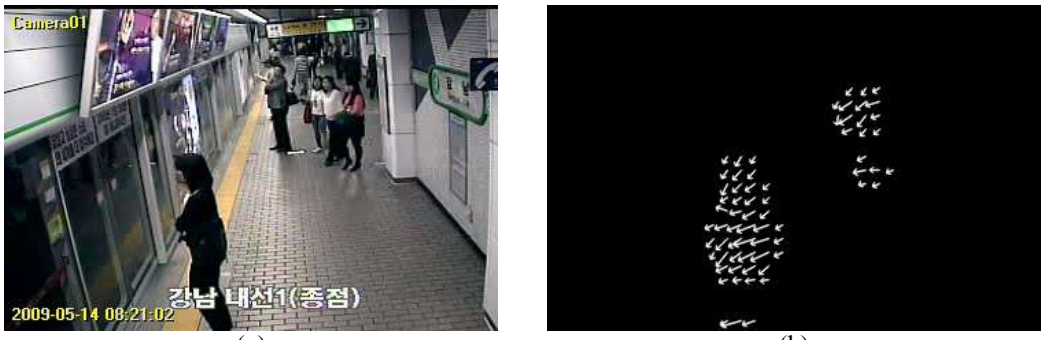
$$J=\begin{bmatrix}J_{11}&J_{12}&J_{13}\\J_{21}&J_{22}&J_{23}\\J_{31}&J_{32}&J_{33}\end{bmatrix}=\begin{bmatrix}f_x^2&f_xf_y&f_xf_t\\f_yf_x&f_y^2&f_yf_t\\f_tf_x&f_tf_y&f_t^2\end{bmatrix} \tag{4}$$

$E(u,v)$ is minimized by solving its Euler-Lagrange equation (5) given by

$$\begin{aligned}0&=\sum_{j\in R(i)}\frac{u_i-u_j}{h^2}-\frac{1}{\lambda}(J_{11i}u_i+J_{12i}v_i+J_{13i})\\0&=\sum_{j\in R(i)}\frac{v_i-v_j}{h^2}-\frac{1}{\lambda}(J_{21i}u_i+J_{22i}v_i+J_{23i})\end{aligned} \tag{5}$$

where h denotes the size of the rectangular pixel grid, u_i and v_i are the approximation to u and v at some pixel i respectively. J_{nmi} denotes the $(n,m)_{th}$ component of the structure tensor of pixel i. $R(i)$ represents the set of neighbors of pixel i.

Fig. 5 shows the computed optical flow for the spatially transformed Gangnam subway station image, **Fig. 4-(a)**. The optical flow vector of a pixel is drawn as a vector located at that pixel coordinate.



(a) (b)

Fig. 5. (a) original image (b) optical flow for original image

The computed optical flow denotes the motion of each pixel at specific time *t*. The moving crowd can be detected based on the computed optical flow. In this paper, we measure the area covered by persons moving between $t-\Delta t$ and *t*, in order to estimate the crowd density. To identify regions corresponding to a person moving between $t-\Delta t$ and *t*, we accumulate optical flow for time window Δt. The magnitude of optical flow for pixel (x, y) at time *t*, $I(x, y, t)$ is computed as equation (6)

$$I(x,y,t)=\sqrt{u(x,y)^2+v(x,y)^2} \tag{6}$$

The accumulated optical flow for pixel (x, y), $AO(x, y)$ is computed as equation (7).

$$AO(x,y)=\sum_{k=t-\Delta t}^{t} I(x,y,k) \tag{7}$$

Pixel (x, y) is defined as a motion cell i.e. $M(x, y)=1$, if $AO(x, y)$ exceeds a predefined threshold $T(x, y)$. Otherwise, the cell (x, y) is not a motion cell i.e. $M(x, y)=0$.

$$M(x,y)=\begin{cases}1 & \text{if } AO(x,y)>T(x,y)\\ 0 & \text{Otherwise}\end{cases} \tag{8}$$

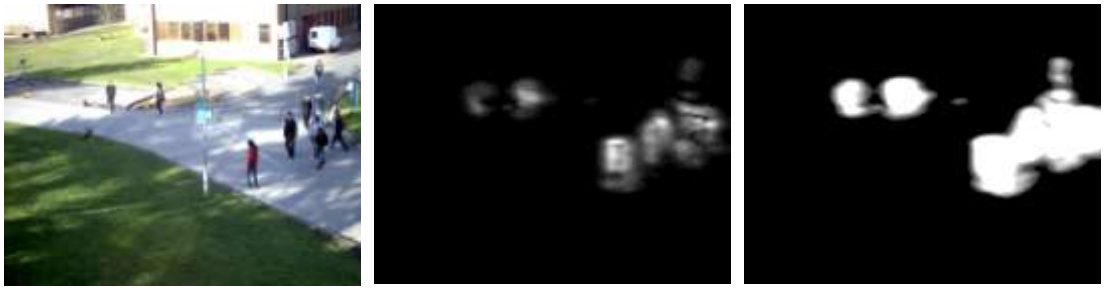
The threshold is set from the computed optical flow as follows.

$$T(x,y)=\alpha\frac{1}{F}\sum_{f=1}^{F}\left[\frac{1}{NM}\sum_{x=1}^{N}\sum_{y=1}^{M}AO(x,y)\right], 0\le\alpha\le 1 \tag{9}$$

where α is constant, F is the number of frames in the training set.

The total number of motion cells in a frame becomes a feature "moving area" denoting an area covered by persons moving between $t-\Delta t$ and t. **Fig. 6-(a)** shows a PETS 2009 original image. **Fig. 6-(b)** shows the result of computed optical flow from **Fig. 6-(a)**. **Fig. 6-(c)** shows

the accumulated optical flow during 5 frames. The grey level of a pixel (x, y) in **Fig. 6 (c)** denotes $AO(x, y)$. The moving area becomes the area of the white region in **Fig. 6(c)**.

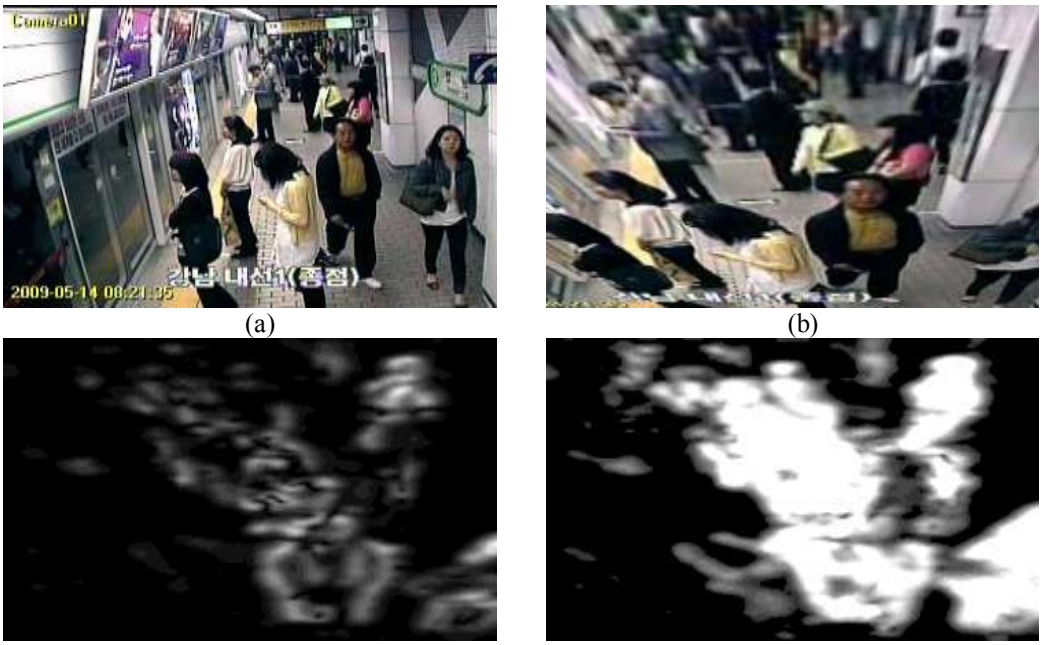


(a) (b) (c)

Fig. 6. (a) Original image (b) Optical flow (c) AO (accumulated optical flow)

Notice that the crowded area of the right part of the image is represented as a wide white area in the AO image. The less crowded area in the top left part of the image is represented as a small white area in the AO image.

Fig. 7 shows the process of computing the accumulated optical flow from the platform image sequence. **Fig. 7-(a)** shows the original image of Gangnam subway station. **Fig. 7-(b)** shows the spatially transformed image from extracted ROI image. **Fig. 7-(c)** shows the image of the computed optical flow from **Fig. 7-(b)**. **Fig. 7-(d)** shows the accumulated optical flow during 5 frames.



(a) (b)

(c) (d)

Fig. 7. (a) Original image (b) Result of spatially transformed image

(c) Optical flow (d) Accumulated optical flow

The relationship between the moving area and the crowd density is shown in **Fig. 8**. In this figure, the moving area is normalized to a real number between 0 and 1, by equation (10).

$$\text{Normalized Moving Area} = \frac{\text{Number of Motion Cells}}{\text{Number of Total Pixels}} \tag{10}$$

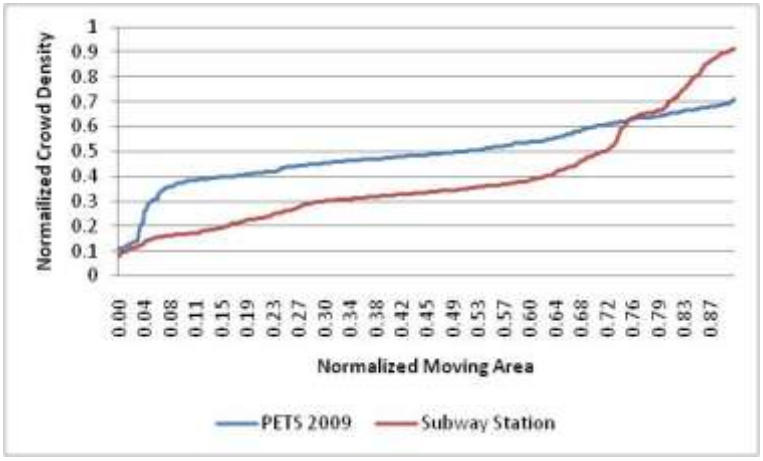


Fig. 8. Relationship between the crowd density and the moving area

The vertical axis of **Fig. 8** denotes the normalized crowd density. The crowd density of each frame is measured by manually counting the people in the frame. **Fig. 8** shows that the moving area increases as the crowd density increases, in both test image sequences. For the subway station image, the small increase of the moving area in the range of 0.7-0.9 results in the rapid increase of crowd density. The reason is that in a subway station, the moving area does not increase much, due to motion interference among persons when it is crowded. Thus the moving area is a good feature to estimate the crowd density.

Another feature is a distribution characteristics of the motion cells. As it is more crowded, people will be spread evenly across the image plane. The distribution of motion cells is measured using random variables cm and rm.

cm [i] = number of motion cells in the i^{th} column of pixels. i=0,1,2,...,M-1
rm [j] = number of motion cells in the j^{th} row of pixels. j=0,1,2,…,N-1

where M and N are the total number of pixel columns and rows, respectively.

The scene is regarded as more crowded, if cm and rm are distributed uniformly. The sample standard deviations of cm and rm are computed as S_{cm} , S_{rm} , respectively, as follows

$$S_{cm} = \sqrt{\frac{1}{M-1}\sum_{i=1}^{M}\left(cm[i]-\overline{cm}\right)^2} = \sqrt{\frac{M\sum_{i=1}^{M}cm[i]^2-\left(\sum_{i=1}^{M}cm[i]\right)^2}{M(M-1)}} \tag{11}$$

$$S_{rm} = \sqrt{\frac{1}{N-1}\sum_{i=1}^{N}\left(rm[i]-\overline{rm}\right)^2} = \sqrt{\frac{N\sum_{i=1}^{N} rm[i]^2 - \left(\sum_{i=1}^{N} rm[i]\right)^2}{N(N-1)}} \tag{12}$$

$$\overline{cm} = \frac{1}{M}\sum_{i=1}^{M} cm[i] \ , \ \overline{rm} = \frac{1}{N}\sum_{i=1}^{N} rm[i] \tag{13}$$

where $\overline{cm}$, $\overline{rm}$ are sample means of cm and rm, respectively.

The moving area, however, increases in accordance to any increase in velocity. For example, similar moving areas are extracted from a single person running and from two people walking. To solve the problem in people counting that arises from moving areas when measuring crowd density, we suggest the use of contrast as an independent feature from moving area. Since contrast indicates the differences between pixels that can be measured independently of crowd movements, it prevents problems that may appear in moving areas and detects stationary crowds.

2.3 Contrast

In very crowded situations, such as rush hour in a subway station, there is no motion of passengers, especially in staircase bottlenecks. Also, there are persons in the scene without any movement. To detect stationary people, we adopt a gray level dependence matrix method [7]. This method is selected, as we assumed that the crowd formulate texture, since the appearance of persons at a constant distance can be regarded as texture. Thus, low density crowds tend to present coarse textures, while dense crowds tend to present fine textures. The gray level dependence matrix represents the second-order joint conditional probability density functions, $f(i, j \mid d, \theta)$. Each $f(i, j \mid d, \theta)$ is the probability of the pair of grey levels (i, j) occurring in a pair of pixels of the image, given that these pixels are separated by a distance d along the direction θ.

$$f(i, j \mid d, \theta) = \frac{\text{number of pixel pairs with gray level } (i,j)}{\text{total number of pixel pairs separated by distance } d \text{ along direction } \theta} \tag{14}$$

To avoid computational burden, we restrict these parameters, d and θ, to a limited number of values. Only four angles (0° , 45° , 90° , 135°), and only one distance (one pixel width), are used in this paper.

A gray level dependence matrix for fine texture tends to be more uniformly dispersed than a gray level dependence matrix for coarse texture. In order to measure the relative spread of a gray level dependence matrix, we use the contrast measure of the spread indicators proposed by Haralick, as shown in equation (15).

$$S_c(d,\theta) = \sum_{i=0}^{L-1}\sum_{j=0}^{L-1}(i-j)^2 f(i, j \mid d, \theta), \ \ d = 1, \theta = 0^o, 45^o, 90^o, 135^o \tag{15}$$

Finally, the gray level dependence matrix is summed and normalized.

$$S = \sum_{\theta=0^o}^{135^o} S_c(1,\theta) / S_{\max} \tag{16}$$

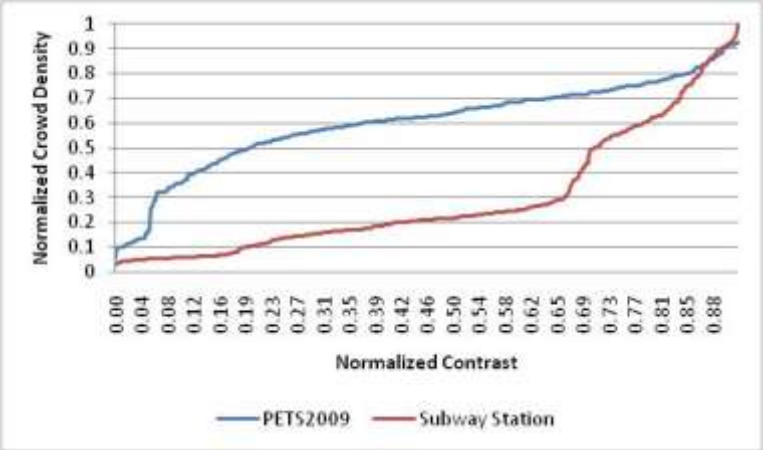


Fig. 9. Relationship between the crowd density and the normalized contrast

Fig. 9 shows the relationship between the crowd density and the contrast for two test image sequences. The crowd density in images of PETS2009 increases gradually, while the crowd density in images of Gangnam subway station increases rapidly.

3. Neural Network Model

A neural network is an information processing paradigm that is inspired by the way biological nervous systems, such as the brain, process information. Neural networks are sometimes called machine learning algorithms, because changing of its connection weights causes the network to learn the solution to a problem. The strength of connection between the neurons is stored as a weight-value for the specific connection. The system learns new knowledge by adjusting these connection weights.

The backpropagation algorithm used in this paper proves to be highly successful in the training of a multilayer neural network. Backpropagation is a gradient descent algorithm, in which the network weights are moved along the negative of the gradient of the performance function. The training begins with random weights, and the goal is to adjust them so that the error will be minimal.

Using smaller network size does not guarantee to converge to the best solution, but using larger network size requires large computational demand and may be over-fitting. In this paper, a multilayer network is proposed, since the decision region for each class is not single hyperplane but convex region. In order to reduce the over fitting problem, we let hidden layer be 1. The number of nodes in the hidden layer is carefully decided as 6 in trial and error method.

A neural network with 2 input units, 6 hidden units and 1 output unit is used. The input nodes are connected to the extracted features i.e. moving area and contrast information. The value of the output node is a real number between 0 and 1. The output value is uniformly quantized to 5 levels as: Very Low (0~0.2), Low (0.2~0.4), Medium (0.4~0.6), High (0.6~0.8), and Very High (0.8~1.0). **Fig. 10** shows the designed multilayer perceptron model.

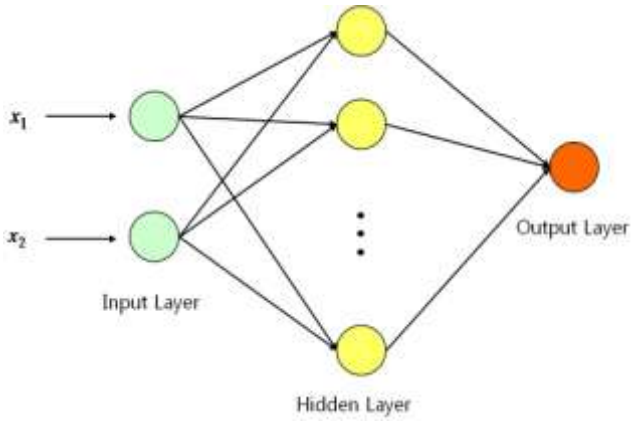


Fig. 10. Our proposed 2-layer perceptron for crowd density measurement

The activation function is in the form of $f_s(x) = 1/(1+\exp(-x))$. This function is smooth, continuous, and differentiable. It has an absolute upper and a lower limit. The sigmoid function is one of the most widely used activation functions, and is illustrated in **Fig. 11**:

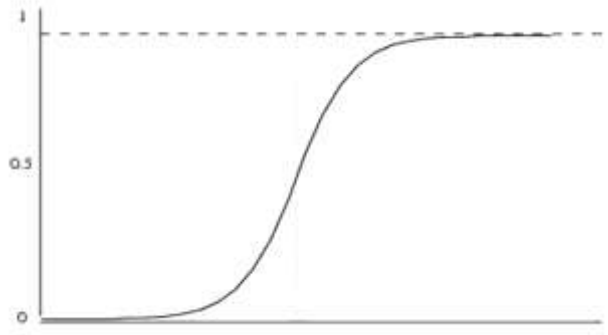


Fig. 11. Sigmoid function

The following notation is used. Let u_{ji} denote the weight of the i^{th} input node to the j^{th} neuron of the hidden layer; let w_{kj} denote the weight from the j^{th} hidden neuron to the k^{th} neuron of the output layer. Let $\{(x^{(1)},d^{(1)}),(x^{(1)},d^{(1)}),\cdots,(x^{(p)},d^{(p)})\}$ be a set of p training patterns, where $x^{(i)} \in R^2$ is the i^{th} input vector in the 2-dimensional feature space, and $d^{(i)} \in [0,1]$ is a desired output value for the i^{th} input vector. Let $o^{(i)} \in [0,1]$ be an actual value of output node for i^{th} input vector.

A Sum squared-error function, which is given by

$$E = \frac{1}{2}\sum_{t=1}^{p}\left\|o^{(i)} - d^{(i)}\right\|^2 \tag{17}$$

is selected to be a cost function. The learning algorithm strives to minimize the learning error defined by the cost function.

The network acts as an approximating function $F(U,W,x^{(i)})$, with two sets of fixed weights:

$$U = \begin{pmatrix} u_{11} & u_{12} \\ u_{21} & u_{22} \\ \vdots & \vdots \\ u_{61} & u_{62} \end{pmatrix}, \; W = \begin{pmatrix} w_{11} & w_{12} & \cdots & w_{16} \end{pmatrix} \tag{18}$$

The output of the neuron of the output layer is given by

$$o_k = F(U, W, x^{(i)}) = F_s(Wv_k) = f_s\left(\sum_{j=1}^{6} w_{1j} f_s\left(\sum_{i=1}^{2} u_{ji} x_i\right)\right), \; 1 \le i \le 2, \; 1 \le j \le 6 \tag{19}$$

In equation (19), v_k denotes a vector of outputs of hidden nodes. Here, v_k is written in matrix form i.e.

$$v_k = F_s(Ux^{(i)}) \;\; \text{where } F_s\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} f_s(x_1) \\ f_s(x_2) \end{pmatrix} \tag{20}$$

4. Experimental Results

In this section, experimental results of crowd estimation by proposed multilayer neural network are presented. Two test image sequences are selected. One is from PETS 2009, as an outdoor image sequence; and the other is an actual Gangnam subway station platform image sequence, as an indoor image sequence. **Fig. 12** shows the use of the PETS 2009 and Gangnam subway station platform image sequences. For each image sequence, 5 classes of crowd density are shown.

(a) (b) (c) (d) (e)

Fig. 12. 5 Crowd density classes; the top row shows images of outdoors (PETS 2009), the bottom row show images of indoors (the platform of Gangnam subway station in Seoul) (a)Very Low (b)Low (c) Medium (d) High (e) Very High

The PETS 2009 dataset consists of 682 frames of 768 X 576 images. The number of people appearing in each image is manually counted. According to the number of people, the crowd

density of each image is labeled as one of 5 classes. The rule used for classifying the crowd densities of PETS 2009 dataset is as follows: very low density (from 0 to 8 people), low density (from 9 to 16 people), medium density (from 17 to 24 people), high density (from 25 to 32 people), and very high density (more than 33 people).

The numbers of frames for each class in PETS2009 are as follows: 16 frames of very low density, 146 frames of low density, 302 frames of medium density, 193 frames of high density, and 25 frames of very high density.

The platform image sequence consists of 552 frames of 360 X 240 images. Each frame of the platform image sequence is labeled similarly. Firstly, we count the number of people appearing in each frame, and labeled crowd density using the following rule: very low density (from 0 to 15 people), low density (from 16 to 30 people), medium density (from 31 to 45 people), high density (from 46 to 60 people), and very high density (more than 61 people).

The numbers of frames for each class in the subway station are as follows: 401 images of very low density, 65 images of low density, 27 images of medium density, 31 images of high density, and 28 images of very high density.

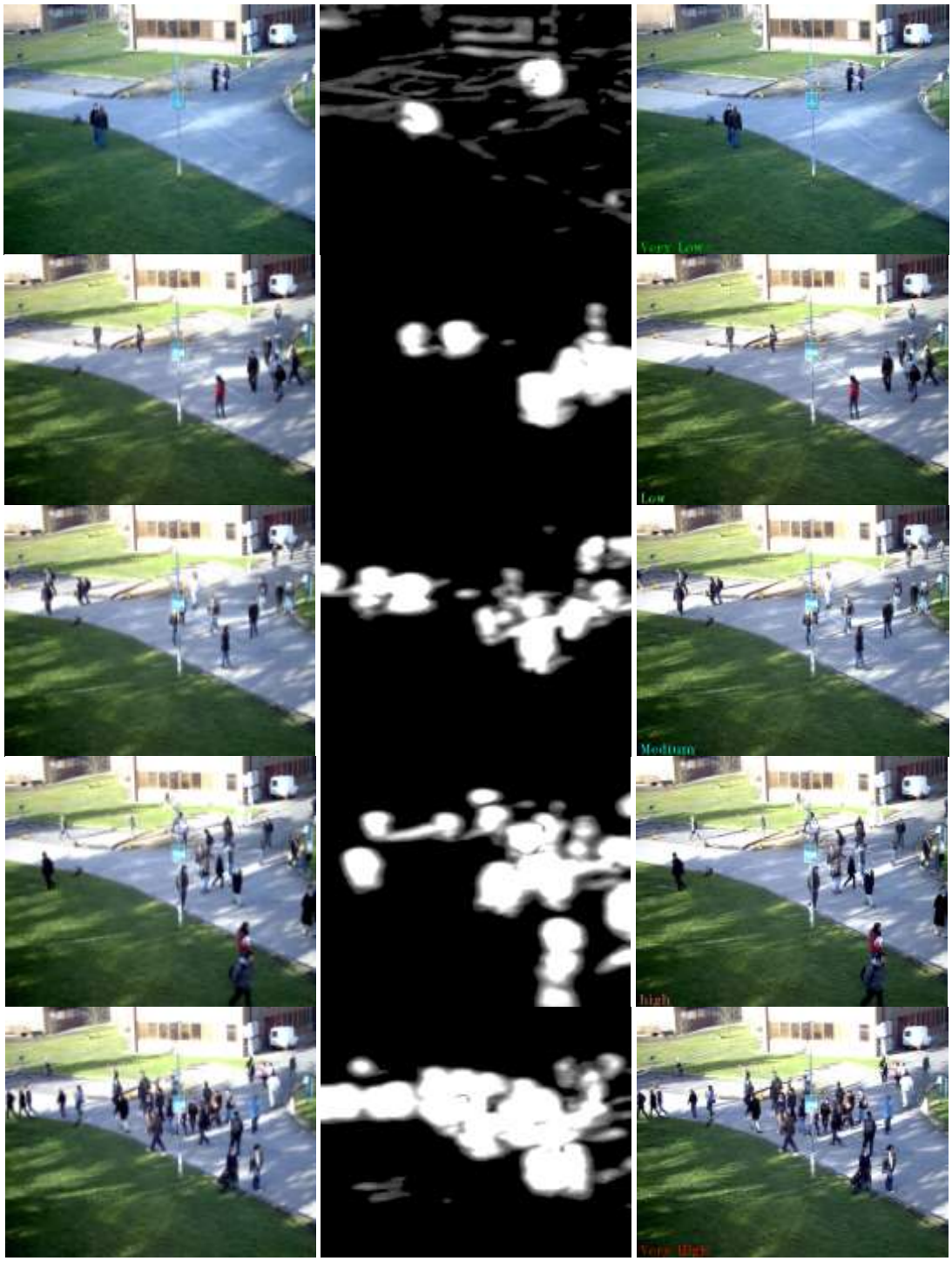
Approximately one half of the total frames in each image sequence is used as a training set to train the neural networks, while the other half is used as a test set to measure the performance of the proposed model. For training, the desired output value of a training data, $x^{(i)}$, is computed as the ratio of the number of people in $x^{(i)}$, to the maximum number of people in the given image sequence. **Table 1** shows the number of frames in the training set and test set for each class.

Table 1. The number of frames in each class of crowd density

Dataset	Group	Very Low	Low	Medium	High	Very High	Total
PETS 2009	Training set	9	74	152	97	14	346
	Test set	7	72	150	96	11	336
Subway Station	Training set	201	33	14	16	15	279
	Test set	200	32	13	15	13	273

The crowd density is measured in real time, and classified into 5 classes. The class of the estimated crowd density is displayed in the captured image in real time.

In **Fig. 13**, we show 5 frames of PETS2009 dataset in the first column, the accumulated optical flow of each pixel in the second column, which is shown in the grey level in the image plane, and images labeled with estimated crowd densities in the third column. The optical flow is accumulated for 5 frames. Notice that the white region, which is a set of pixels with a large value of accumulated optical flows, becomes larger, as crowd density increases. The result of the estimated crowd density level is shown as the text in the bottom left corner of the image in the third column.



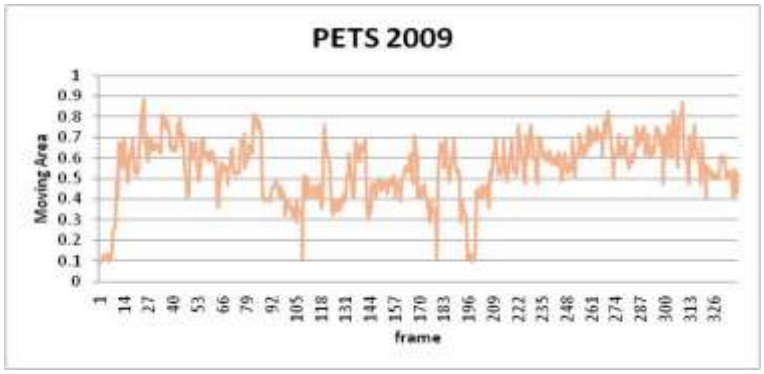
(a) (b) (c)

Fig. 13. Estimation of crowd density by multilayer neural network for PETS 2009 sequence

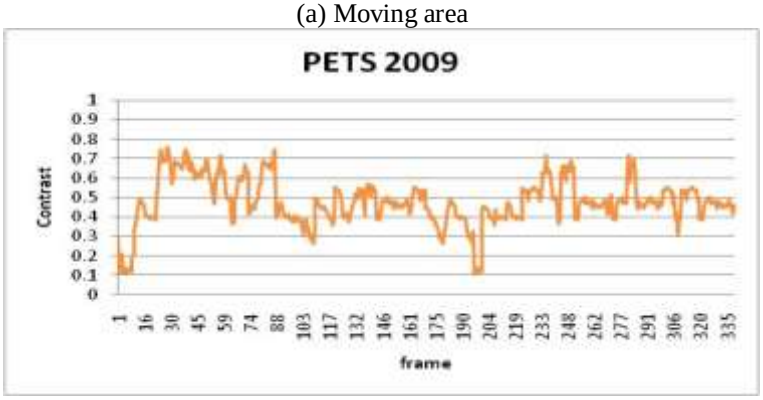
(a) Original image frames (b) Accumulated optical flow vectors

(c) Estimated crowd density as one of 5 classes

Fig. 14 shows the changes in moving area and contrast, during 336 frames in the PETS 2009 dataset. The normalized motion cell ratio changes almost similarly with the contrast. **Fig. 15** shows the estimated crowd density, and manually measured crowd density, for each frame of the image sequence. **Fig. 15** shows the change of the estimated crowd density according to the frame number, and this clearly shows the variation of crowd density in PETS 2009 image sequence.



(a) Moving area



(b) Contrast

Fig. 14. Change of moving area and contrast in image sequence (PETS 2009)

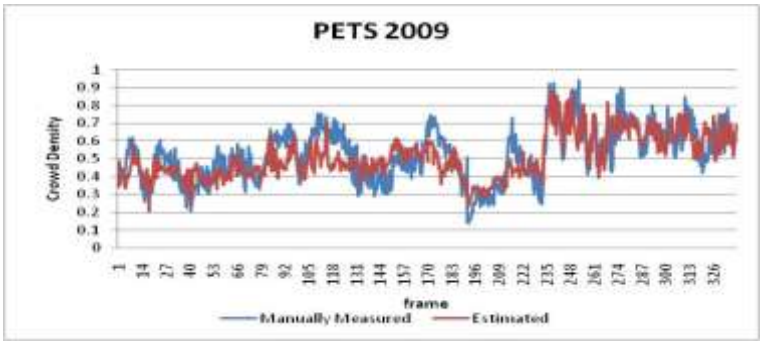


Fig. 15. Estimated crowd density (red line) and manually measured crowd density (blue line) for each frame of PETS 2009 image sequence

In **Fig. 16**, we show 5 frames of the platform image sequence in the first column, the

accumulated optical flow of each pixel in the second column, which is shown in the grey level in the image plane, and images labeled with the estimated crowd densities in the third column.

(a) (b) (c)

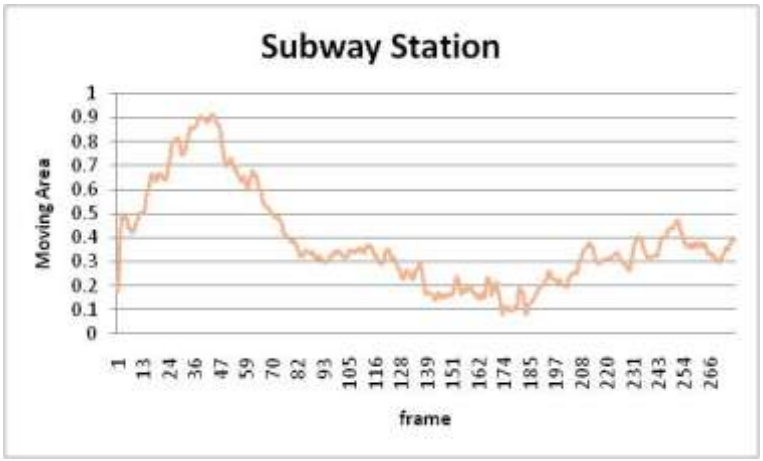
Fig. 16. Estimation of crowd density by multilayer neural network for platform sequence
(a) Original image frames (b) Accumulated optical flow vectors
(c) Estimated crowd density as one of 5 classes

The optical flow vector is accumulated for 5 frames. The accumulated optical flow vector is

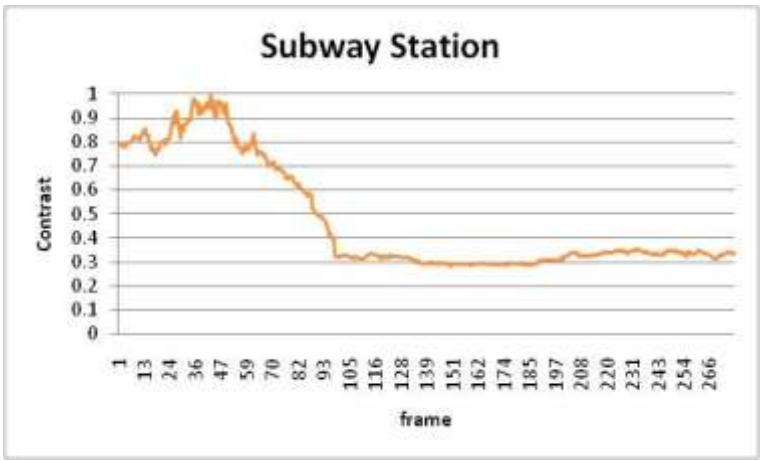
plotted only for the ROI region, which is spatially transformed. The image plane for the "very high" class is filled with white regions. The result of the estimated crowd density is shown as text in the top left corner of the image in the third column.

Fig. 17 shows the changes in moving area and contrast during 273 frames in the platform image sequence. The normalized motion cell ratio changes almost similarly with the contrast.

Fig. 18 shows the estimated crowd density and manually measured crowd density for each frame of the image sequence. Notice that the similarity of changes in the moving area with changes in the crowd density for the crowded frames (from frame 1 to frame 91). For the relatively uncrowded frames (from frame 92 to frame 273), the changes in contrast are more similar to changes in crowd density. **Fig. 18** shows the change of the estimated crowd density with respect to the frame number, and this clearly shows the variation of crowd density in the platform image sequence.



(a) Moving area



(b) Contrast

Fig. 17. Change of moving area and contrast in platform image sequence

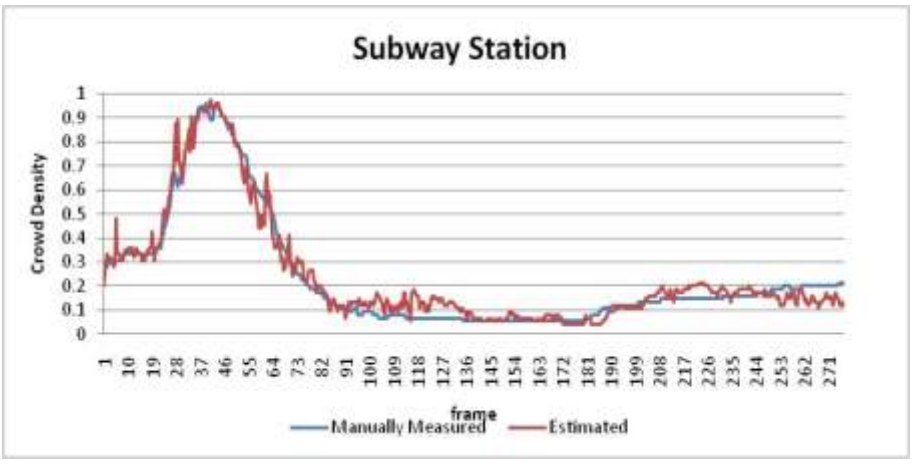


Fig. 18. Estimated crowd density (red line) and manually measured crowd density(blue line) for each frame of platform image sequence

Table 2 shows the performance of the crowd density estimation for 2 image sequences. Each row of the table represents a set of test image frames with a specified density level. The crowd density of each image in this set is estimated. Then the test images are classified, according to the estimated densities. The ratio of classified test images in a class to the total images in the test set is computed. For example, the first row of PETS2009 image sequence is a set of frames manually labeled "very low". The images in this set are classified either "very low" or "low". The ratio of images classified as "very low", to total images in this set, is 91.8%. The ratio of images classified as "low", to total images in this set, is 8.2%. Thus the correct classification ratio becomes 91.8%. The numbers in the diagonal element of the table indicate the correct classification ratio for each class of images. The best result is 91.8% of correct classification obtained for the "very low" class and the worst result is 71.5% of correct classification obtained for the "high" class. It is possible to observe that all miss-classified images are allocated only to a neighbor class of the correct one. Some of miss classifications are expected since the borders between the crowd density classes are very ambiguous. For instance, an image with 24 people belongs to "medium" class, but it can be easily classified to "high" class.

Table 2. Results of crowd density estimation in PETS 2009 and platform image sequences

Dataset	Density Level	Very Low	Low	Medium	High	Very High	Correct Classification
PETS 2009	Very Low	91.8%	8.2%	-	-	-	91.8%
	Low	8.4%	81.2%	10.4%	-	-	81.2%
	Medium	-	9.8%	83.1%	7.1%	-	83.1%
	High	-	-	14.7%	71.5%	13.8%	71.5%
	Very High	-	-	-	13.4%	86.6%	86.6%
Platform	Very Low	96.9%	3.1%	-	-	-	96.9%
	Low	8.8%	86.8%	4.4%	-	-	86.8%
	Medium	-	15.3%	72.8%	11.9%	-	72.8%
	High	-	-	9.6%	74.3%	16.1%	74.3%
	Very High	-	-	-	6.2%	93.8%	93.8%

In the platform image sequence, the estimation performance (correct classification ratio) drops, when the class of an image is "medium" or "high". The reason for low accuracy is that the moving area in **Fig. 16** does not increase proportionally to the crowd density. On the other hand, in both sequences, the estimation performs well for the extreme classes i.e. "very low" and "very high". The best result is 96.9% of correct classification obtained for the "very low" class and the worst result is 72.8% of correct classification obtained for the "medium" class. All miss-classified images are allocated only to a neighbor class of the correct one.

This result can be considered very satisfactory, when it is observed that variances of estimation results for each class are very small, and their means are very close to the actual values. Also, it shows that our proposed method is effective in handling the outdoor environments with dense crowd and various person sizes, and indoor environments with significant body occlusion with stationary crowd problems.

5. Conclusions

This paper presents a multilayer neural network model, for estimation of crowd density from captured image frames. We also develop two effective features that do not require counting people, thus segmentation or edge detection of shadow regions becomes irrelevant. The major clue is optical flow, and is used to measure moving area in short time periods. Also, the coarseness of texture is measured as a complementary feature to estimate stationary crowd density. The neural network is trained with the PETS2009 dataset, and actual captured platform image sequences. Our own training sets are constructed for both image sequences. We labeled each frame as one of 5 classes, by counting the number of people appearing in the frame. The estimation accuracy for each class is measured, using test images selected from two image sequences. The result shows that the model can estimate crowd density successfully in indoor and outdoor image sequences. Another advantage of this technique is its independence from the apparent shape of individual pedestrians. Our proposed neural network model for estimation of crowd density facilitates crowd monitoring or analyzing crowd behavior in public areas in real time. Further research needs to be performed to develop additional features, to increase estimation accuracy.

References

[1] Sergio A. Velastin, Boghos A. Boghossian and Maria Aclicia Vicencio-Silva, "A motion-based image processing system for detecting potentially dangerous situations in underground railway stations", *Transportation Research Part c: Emerging Technologies,* vol.14, no.2, pp. 96-113, Apr. 2006. Article (CrossRef Link)

[2] Ruihua Ma, Liyuan Li, Weimin Huang and Qi Tian, "On pixel count based crowd density estimation for visual surveillance", in *Proc. of IEEE Conf. on Cybernetics and Intelligent Systems,* pp.170-173, Dec.2004. Article (CrossRef Link)

[3] Dang Kong, Doug Gray and Hai Tao, "A viewpoint invariant approach for crowd counting", in *Proc. of 18th Int. Conf. on Pattern Recognition*, pp.1187-1190, 2006. Article (CrossRef Link)

[4] Dang Kong, Doug Gray and Hai Tao, "Counting Pedestrians in crowds using viewpoint invariant training", in *Proc. of British Machine Vision Conf.*, 2005. Article (CrossRef Link)

[5] Beibei Zhan, Dorothy N. Monekosso, Paolo Remagnino, Sergio A. Velastin and Li-Qun Xu, "Crowd analysis: a survey", *Machine Vision and Applications,* vol.19, pp.345-357, Apr.2008. Article (CrossRef Link)

[6] A. N. Marana, L. F. Costa, R. A. Lotufo and Sergio A. Velastin, "Estimating crowd density with Minkowski fractal dimension", in *Proc. of IEEE Int. Conf. on Acoustics, Speech, and Signal*

Processing, pp.3521-3524, Mar.1999. Article (CrossRef Link)
[7] A. N. Marana, Sergio A. Velastin, L. F. Costa and R. A. Lotufo, "Automatic estimation of crowd density using texture", *Safety Science*, vol.28, no.3, pp.165-175, Apr.1998. Article (CrossRef Link)
[8] T. K. An and M. H. Kim, "Context-aware Video Surveillance System", *Journal of Electrical Engineering and Technology,* vol.7, no.1, pp.115-123, Jan.2012. Article (CrossRef Link)
[9] K. Y. Eom, J. Y. Jung, and M. H. Kim, "A heuristic search-based motion correspondence algorithm using fuzzy clustering", *International Journal of Control, Automation and Systems,* vol.10, no.3, pp.594-602, 2012. Article (CrossRef Link)
[10] Rafael C. Gonzalez and Richard E. Woods, *Digital Image Processing,* 2nd Edition, Prentice Hall, 2002.
[11] J. Y. Jung and M. H. Kim, "Motion estimation of lips in pronouncing korean vowels based on fuzzy constraint line clustering", in *Proc. of IEEE Int. Conf. on Image Processing,* pp.507-510, Sep. 1996. Article (CrossRef Link)
[12] H. J. Kim, H. S. Kang, S. H. Lee and M. H. Kim, "A study on fuzzy constraint line clustering for optical flow estimation", *Journal of The Institute of Electronics Engineers of Korea,* vol.31, no.9, pp.1403-1411, Sep.1994.
[13] G. J. Kim, T. K. Ahn, K. Y. Eom, J. Y. Jung and M. H. Kim, "Automated Measurement of Crowd Density Based on Edge Detection and Optical Flow", in *Proc. 2nd Int. Conf. on Industrial Mechatronics and Automation*, pp.553-556, May. 2010. Article (CrossRef Link)
[14] Andres Bruhn, Joachim Weickert, Christian Feddern, Timo Kohlberger and Christoph Schnorr, "Real-Time optic flow computation with variational methods", in *Computer Analysis of Images and Patterns,* pp.222-229, 2003. Article (CrossRef Link)

Gyu Jin Kim received the B.S. degree in Applied Mathematics from Sejong University in 2009, and the M.S. degree in Electrical and Computer Engineering from Sungkyunkwan University, Seoul, Korea in 2011. His research interests include pattern recognition, machine learning, computer vision, and artificial intelligence.

Tae Ki An received the B.S. degree and the M. S. degree in Electronic Engineering from Kyungpook National University, Daegu, Korea in 1994 and 1996. The Ph.D. degree in Electrical and Computer Engineering from Sungkyunkwan University, Seoul, Korea in 2011. He is a Chief Researcher in Korea Railroad Research Institute. His research interests are artificial intelligence, pattern recognition, and video analysis.

Moon Hyun Kim received the B.S. degree in Electronic Engineering from Seoul National University in 1978, the M.S. degree in Electrical Engineering from KAIST, Korea, in 1980, and the Ph.D. degree in Computer Engineering from the University of Southern California in 1988. From 1980 to 1983, he was a Research Engineer at the Daewoo Heavy Industries Co., Seoul. He joined the School of Information and Communication Engineering, Sungkyunkwan University, Seoul, Korea in 1988, where he is currently a Professor. In 1995, he was a Visiting Scientist at the IBM Almaden Research Center, San Jose, California. In 1997, he was a Visiting Professor at the Signal Processing Laboratory of Princeton University, Princeton, New Jersey. His research interests include computer vision, pattern recognition, and artificial intelligence.