EXTRACTING 3D SEMANTIC INFORMATION FROM VIDEO SURVEILLANCE SYSTEM USING DEEP LEARNING

J. S. Zhang [1,*], J. Cao [1], B. Mao [1], D. Q. Shen [2]

[1] Nanjing University of Finance & Economics, College of Information Engineering, Collaborative Innovation Center for Modern Grain Circulation and Safety, Jiangsu Key Laboratory of Modern Logistics, Nanjing, 210023, China - jianshu.zhang@foxmail.com, caojie690929@163.com, maoboo@gmail.com
[2] Nanjing University of Science & Technology, Nanjing, 210094, China - sdq871206@163.com

Commission III Urban Sensing and Mobility

KEY WORDS: 3-D space, Camera calibration, Target recognition, Target tracking

ABSTRACT:

At present, intelligent video analysis technology has been widely used in various fields. Object tracking is one of the important part of intelligent video surveillance, but the traditional target tracking technology based on the pixel coordinate system in images still exists some unavoidable problems. Target tracking based on pixel can’t reflect the real position information of targets, and it is difficult to track objects across scenes. Based on the analysis of Zhengyou Zhang's camera calibration method, this paper presents a method of target tracking based on the target's space coordinate system after converting the 2-D coordinate of the target into 3-D coordinate. It can be seen from the experimental results: Our method can restore the real position change information of targets well, and can also accurately get the trajectory of the target in space.

1. INTRODUCTION

In recent years, with the popularity of network cameras for security, intelligent video surveillance technology has rapidly become a research focus. Target detection, target recognition and target tracking are three main parts in intelligent video surveillance technology. Target tracking is used to determine the location of targets which we are interested in video sequences. It is a basic problem in the field of computer vision, and has wide application value.

The traditional target tracking technology is to record moving tracks of targets in 2-D images. This method is easy to implement and can record the trajectory of targets in current scene. However, this method is limited to the pixel space, and can’t reflect the position change information of targets in the space, and it’s hard to track targets across scenes.

On the basis of studying the principle of camera imaging and camera calibration, this paper presents an automatic conversion method between 2-D point coordinate and 3-D point coordinate, maps the coordinates of the target in the 2-D images into real space, and then obtain the historical trajectory of targets in space.

2. RELATED WORK

2.1 Camera Calibration Technology

A camera is a mapping between 3-D space and 2-D images. The relationship between these two spaces is determined by the geometrical model of the camera, which is commonly called the camera parameter. The camera model is a geometric abstraction of the real camera, and the camera imaging process is the transformation of the space point of photography. Linear model (aperture imaging model) is one of the commonly used camera imaging models.

The purpose of camera calibration is to determine the geometrical and optical properties within the camera (internal parameters of the camera) and the coordinate relationship of camera in 3-D world (external parameters of the camera) by using the coordinates of the feature point (X, Y, Z) of the given 3-D space object and the image coordinates (u, v) of its 2-D image space. Faig (1975) took into account the various factors in the camera imaging process, and uses at least 17 parameters to describe the constraint relationship between the image and the 3D object space. In 1971, Abdel-Aziz and Karara proposed the direct linear transformation method: the parameters of the camera model can be obtained by solving linear equation. The camera calibration method based on perspective transformation matrix is the use of linear method to solve the various elements of the perspective transformation matrix (Luh J Y S, Klaasen J A. 1985). In 1986, Tsai presented a two-step calibration method based on radial constraints, which has high accuracy rate, but also has high requirements for equipment. Martins et al. first proposed the two plane model. The advantage of this method is that it can use linear method to solve the related parameters. The disadvantage is that a large number of unknown parameters are required to be solved, and there is a tendency for excessive parameterization. In 2000 Zhang Z proposed a new, flexible calibration method. Although this method also used pin-hole model, but this calibration method combines the self-calibration and traditional calibration ideas. Zhengyou Zhang's camera calibration method is one of the most commonly used calibration methods.

2.2 Target Detection and Recognition

Target detection and target recognition are the basis of intelligent video surveillance technology. We can determine whether there

[*] Corresponding author. Tel.:+86-18795853792
E-mail address: jianshu.zhang@foxmail.com

This contribution has been peer-reviewed.
https://doi.org/10.5194/isprs-archives-XLII-3-2257-2018 | © Authors 2018. CC BY 4.0 License.

are targets in video frames, and determine the category and location of targets by target detection and target recognition.

Frame difference method is one of the easiest foreground detection method. In 2006, Migliore et al. improved frame difference method to detect the moving objects in the current scene. Background modeling method based on GMM is a widely used method in the target detection (Zhu Q et al. 2012). In 2004, Zoran Zivkovic present an adaptive algorithm using Gaussian mixture probability density.

Since 2012, deep learning technology has made a breakthrough in the field of target recognition. At present, the main depth learning models are: Auto-encoder (Zivkovic Z et al. 2006), Restricted Boltzmann Machine (RBM) (Smolensky P. 1986), Deep Belief Nets (DBN) (Hinton G E et al. 2006) and Convolutional Neural Networks (CNN) (Lecun Y et al. 1998). In 2012, Professor Hinton and his student Krizhevsk using GPU combined with deep learning won the ImageNet champion that year (Krizhevsky A et al. 2012). In 2014, Ross Girshick proposed the R-CNN (Girshick R et al. 2014) for target recognition and in 2015 he proposed two accelerated versions of R-CNN: Fast R-CNN (Girshick R. 2015) and Faster R-CNN (Ren S et al. 2016). Faster R-CNN makes the Region Proposal Networks (RPN) and the detection network share the convolution feature of the whole picture, which greatly improves the efficiency of target recognition. This makes real-time target recognition possible.

3. METHODOLOGY

A 2D point is denoted by $\mathbf{m} = [\mathrm{u}, \mathrm{v}]^{\mathrm{T}}$. A 3D point is denoted by $\mathbf{M} = [\mathrm{X}, \mathrm{Y}, \mathrm{Z}]^{\mathrm{T}}$. He use $\tilde{\mathrm{x}}$ to denote the augmented vector by adding 1 as the last element: $\widetilde{\mathbf{m}} = [\mathrm{u}, \mathrm{v}, 1]^{\mathrm{T}}$ and $\widetilde{\mathbf{M}} = [\mathrm{X}, \mathrm{Y}, \mathrm{Z}, 1]^{\mathrm{T}}$. The relationship between a 3D point M and its image projection m is given by (1):

$$\mathrm{s}\widetilde{\mathbf{m}} = \mathbf{A}[\mathbf{R}, \mathbf{t}]\widetilde{\mathbf{M}} \tag{1}$$

in which s is an arbitrary scale factor. $[\mathbf{R}, \mathbf{t}]$ is the extrinsic parameter matrix, which is composed of rotation matrix and translation matrix, as in (2):

$$[\mathbf{R}, \mathbf{t}] = \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \end{bmatrix} \tag{2}$$

In which, $[\mathbf{R}]$ is the rotation matrix and $[\mathbf{t}]$ is the translation matrix. The extrinsic parameter matrix is also known as the rotation and translation matrix which relates the world coordinate system and the camera coordinate system. The intrinsic matrix ($\mathbf{A}$), also called the camera matrix is given by (3):

$$\mathbf{A} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \tag{3}$$

with (c_x, c_y) the coordinates of the principal point, f_x and f_y the scale factors in image u and v axes.

Substituting $\widetilde{\mathbf{m}} = [\mathrm{u}, \mathrm{v}, 1]^{\mathrm{T}}$, $\widetilde{\mathbf{M}} = [\mathrm{X}, \mathrm{Y}, \mathrm{Z}, 1]^{\mathrm{T}}$ and (2), (3) into (1), we can get (4):

$$\mathrm{s}\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \tag{4}$$

Making s = 1, (4) can be converted to (5):

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f_x r_{11} X + f_x r_{12} Y + f_x r_{13} Z + f_x t_1 + c_x \alpha \\ f_y r_{21} X + f_y r_{22} Y + f_y r_{23} Z + f_y t_2 + c_y \alpha \\ \alpha \end{bmatrix} \tag{5}$$

In which $\alpha = r_{31}X + r_{32}Y + r_{33}Z + t_3$=1. Then we have:

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f_x r_{11} X + f_x r_{12} Y + f_x r_{13} Z + f_x t_1 + c_x \\ f_y r_{21} X + f_y r_{22} Y + f_y r_{23} Z + f_y t_2 + c_y \\ 1 \end{bmatrix} \tag{6}$$

That is:

$$\begin{cases} u = f_x r_{11} X + f_x r_{12} Y + f_x r_{13} Z + f_x t_1 + c_x \\ v = f_y r_{21} X + f_y r_{22} Y + f_y r_{23} Z + f_y t_2 + c_y \end{cases} \tag{7}$$

Since we only consider the targets on the ground, we set Z=0. Simplify (7), we can get (8):

$$\begin{cases} X = \dfrac{f_x r_{12} v - f_y r_{22} u + (f_y r_{22} f_x t_1 - f_x r_{12} f_y t_2 + c_x f_y r_{22} - c_y f_x r_{12})}{f_x r_{12} f_y r_{21} - f_x r_{11} f_y r_{22}} \\ Y = \dfrac{f_x r_{11} v - f_y r_{21} u + (f_y r_{21} f_x t_1 - f_x r_{11} f_y t_2 + c_x f_y r_{21} - c_y f_x r_{11})}{f_y r_{22} f_x r_{11} - f_y r_{21} f_x r_{12}} \end{cases} \tag{8}$$

Setting: $w_1 = \dfrac{f_x r_{12}}{f_x r_{12} f_y r_{21} - f_x r_{11} f_y r_{22}}$, $w_2 = -\dfrac{f_y r_{22}}{f_x r_{12} f_y r_{21} - f_x r_{11} f_y r_{22}}$, $b_1 = \dfrac{f_y r_{22} f_x t_1 - f_x r_{12} f_y t_2 + c_x f_y r_{22} - c_y f_x r_{12}}{f_x r_{12} f_y r_{21} - f_x r_{11} f_y r_{22}}$; $w_3 = \dfrac{f_x r_{11}}{f_y r_{22} f_x r_{11} - f_y r_{21} f_x r_{12}}$, $w_4 = -\dfrac{f_y r_{21}}{f_y r_{22} f_x r_{11} - f_y r_{21} f_x r_{12}}$, $b_2 = \dfrac{f_y r_{21} f_x t_1 - f_x r_{11} f_y t_2 + c_x f_y r_{21} - c_y f_x r_{11}}{f_y r_{22} f_x r_{11} - f_y r_{21} f_x r_{12}}$,
Then equation (8) can be simplified as (9):

$$\begin{cases} X = w_1 u + w_2 v + b_1 \\ Y = w_3 u + w_4 v + b_2 \end{cases} \tag{9}$$

In which: w_1, w_2, w_3, w_4 are weights, and b_1, b_2 are bias terms. These parameters are all constants.

Based on the deduction above, we find that: the relationship between the 2-D point coordinates and the 3-D point coordinates is a linear. We can use Linear Regression method to solve w_1, w_2, w_3, w_4 and b_1, b_2 in (9).

3.1 Coordinate Acquisition

We chose an open space in the laboratory as an experimental scene. We set a point in front of the door as the origin of the space coordinate system. The horizontal direction is X axis, and the vertical direction is Y axis. The size of each floor tile in laboratory is 60cm*60cm. The schematic diagram is shown in Figure 1.

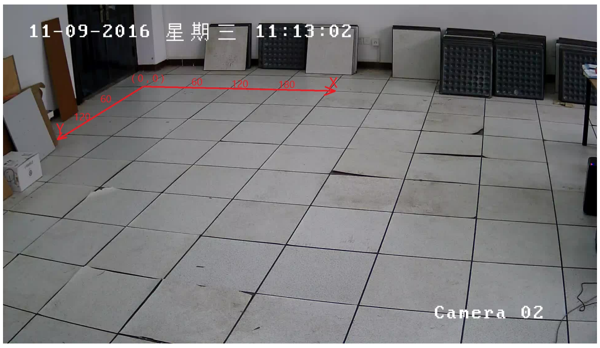


Figure 1. Experimental scene and space coordinate system

Through manual measurement, we can get the coordinates of the floor tiles in the space. For the pixel coordinates corresponding to the spatial coordinates, we can obtain it through target recognition method: Let my classmate stand at the location of the spatial coordinates we measured before, and capture these images. Then we use Faster RCNN (a deep learning framework for target detection and recognition) to recognize the targets and extract the pixel coordinate of each target, as shown in Figure 2.

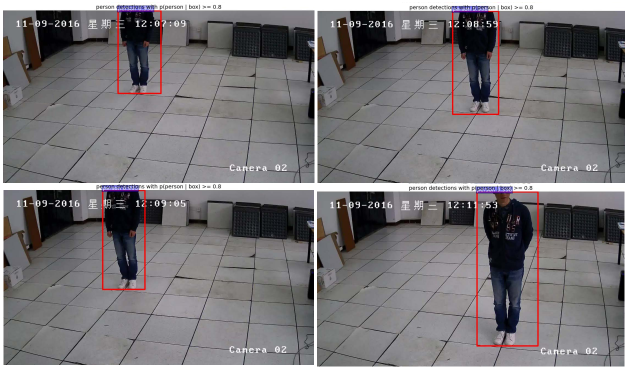

person detections with p(person | box) >= 0.8
11-09-2016 星期三 12:07:09
Camera 02
person detections with p(person | box) >= 0.8
11-09-2016 星期三 12:08:59
Camera 02
person detections with p(person | box) >= 0.8
11-09-2016 星期三 12:09:05
Camera 02
person detections with p(person | box) >= 0.8
11-09-2016 星期三 12:11:53
Camera 02

Figure 2. Pixel coordinate acquisition based on target recognition

3.2 Linear Regression

We obtain 55 sets of one-to-one corresponding 2D point coordinates and 3D point coordinates through the method above, and then use the least squares method to find the parameters (w_1, w_2, w_3, w_4 and b_1, b_2) in the linear regression equation of (9).

The core idea of the least squares method is to select the appropriate weights and bias terms to ensure that the sum of the squares of the deviations of all the fitting results and the actual data is minimal. Taking $X = w_1 u + w_2 v + b_1$ (the equation in the X direction of equation (9)) as an example. Assuming that the sum of squared deviations of all the fitting results and the actual data is S, so S can be expressed as (10):

$$S = \sum_{i=0}^{n} [X_i - (w_1 u_i + w_2 v_i + b_1)]^2 \qquad (10)$$

In which X_i is the abscissa of each 3-D point we measured before, (u_i, v_i) is the corresponding 2-D point coordinates we obtain by target recognition. Therefore, (10) is based on w_1, w_2 and b_1 as independent variables, S as the dependent variable. Then we can get the value of w_1, w_2 and b_1 when S can get to the minimum value, that is solving the following (11):

$$\begin{cases} \frac{\partial S}{\partial w_1} = -2\sum_{i=0}^{n} u_i [X_i - (w_1 u_i + w_2 v_i + b_1)] = 0 \\ \frac{\partial S}{\partial w_2} = -2\sum_{i=0}^{n} v_i [X_i - (w_1 u_i + w_2 v_i + b_1)] = 0 \\ \frac{\partial S}{\partial b_1} = -2\sum_{i=0}^{n} [X_i - (w_1 u_i + w_2 v_i + b_1)] = 0 \end{cases} \qquad (11)$$

3.3 Target tracking in space

First of all, we use the deep learning method to process each frame in the video stream: recognizing the targets in each frame, and obtaining the pixel coordinates of each target. Then use the parameters (w_1, w_2, w_3, w_4 and b_1, b_2) solved from linear regression to transform the coordinates: converting the pixel coordinate (u, v) into space coordinate (X, Y). Finally, we connect the nearest targets in each two adjacent frames in chronological order. This consists the tracks of targets in space. The pseudo-code of the algorithm is as follows:

```
Algorithm 1 Find the track of targets in space
Input: $w_1, w_2, b_1, w_3, w_4, b_2$ (parameter), $T_{mn}$ (consist of m*n tags $(u_i, v_i)$)
Output: $Track$ (the track of targets in space)
for all $(u_i, v_i) \in T_{nm}$ do
    Sort all tags according to the ascending order of i
    $X_i \leftarrow w_1 * u_i + w_2 * v_i + b_1$
    $Y_i \leftarrow w_3 * u_i + w_4 * v_i + b_2$
    $ST_{mn} \leftarrow (X_i, Y_i)$
end for
for all $(X_i, Y_i) \in ST_{mn}$ do
    if $(X_i, Y_i) \in ST_{1n}$ then
        $Start\ list_{1n} \leftarrow ST_{1n}$
        $End\ list_{1n} \leftarrow ST_{1n}$
        $Track \leftarrow [Start\ list_{1n}, End\ list_{1n}]$
    else
        $Start\ list_{in} \leftarrow End\ list_{(i-1)n}$
        for $(X_i, Y_i) \in ST_{in} (i \in [2, m])$ do
            $Dis_{in} \leftarrow$ the distance between $ST_{in}$ and $Start\ list_{in}$
            if $Dis_{in} \leq Threshold$ then
                $End\ list_{in} \leftarrow$ the $ST_{in}$ which makes $Dis_{in}$ minimum
            end if
        end for
        $End\ list_{in} \leftarrow$ the rest of $T_{in}$
        $Track \leftarrow [Start\ list_{in}, End\ list_{in}]$
    end if
end for
```

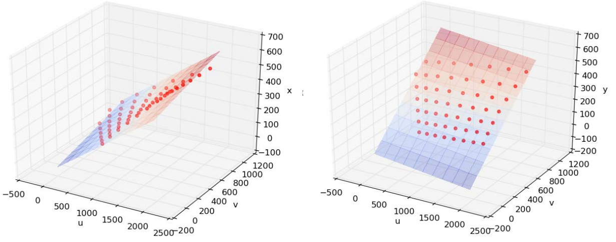
4. EXPERIMENT RESULTS

4.1 The Result of Linear Regression

We use the least squares method mentioned in Section 2.2 to deal with the 55 sets of 2-D coordinate points and 3-D coordinate points we get before. Then solve the parameters: $w_1 = 0.247$, $w_2 = 0.148$, $b_1 = -50.519$, $w_3 = -0.051$, $w_4 = 0.598$, $b_2 = -51.582$. Therefore, the fitted plane equation is as follows:

$$\begin{cases} X = \quad 0.247u + 0.148v - 50.519 \\ Y = -0.051u + 0.598v - 51.582 \end{cases} \qquad (12)$$

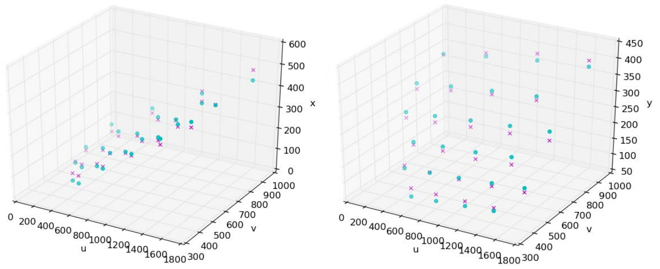
For the abscissa X and the ordinate Y, the plane we obtained and the actual coordinate of each point are shown in Figure 3.



(a) (b)

Figure 3. The fitting results of abscissa and ordinate. (a) is the fitting result of abscissa, and (b) is the fitting result of ordinate

We take another 26 sets of points to test the error of the coordinate transformation. The results are as in Figure 4. In Figure 4, • represents the actual value of the horizontal or vertical coordinates of each point, and × represents the fitted value of the horizontal or vertical coordinates of each point through formula (12).



(a) (b)

Figure 4. The error of the coordinate transformation

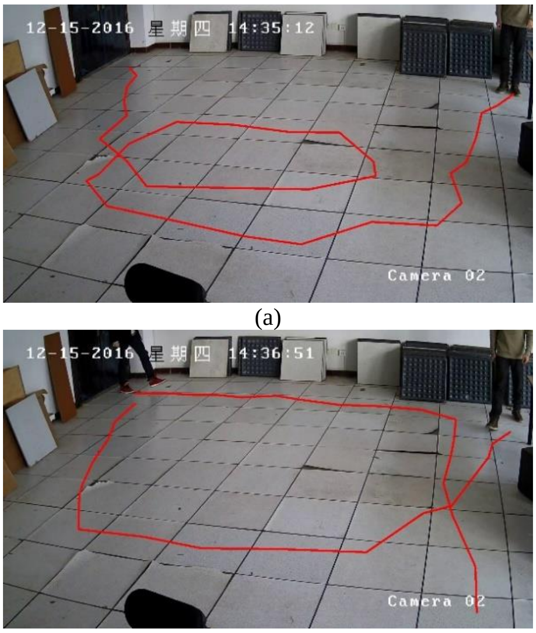
The statistical results of the error between the actual coordinates of the points and the fitted coordinates are shown in Table 1. Taking into account the total area of the scene we tested, the error is within acceptable limits.

Number of test points	Maximum error(cm)	Minimum error(cm)	Average error (cm)
26	30.727	7.846	13.831

Table 1. Statistical results of the error

4.2 The Result of Tracks Drawing

We recorded two videos in the lab with a webcam. Based on the results of the target recognition in each frame of the video, the tracks of targets are drawn during this period of time, as shown in Figure 5.

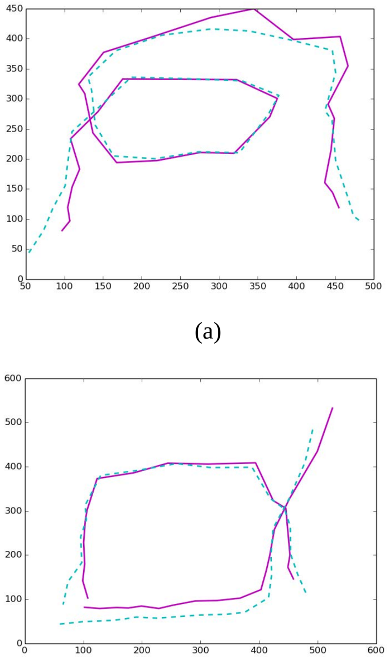


(a)

(b)

Figure 5. Target tracking result based on pixel coordinate system

Through formula (12), we can further obtain the horizontal and vertical coordinates of the target in each frame of the video in the space coordinate system we have determined before. Then draw the tracks of targets in the space coordinate system, as shown in Figure 6.



(a)

(b)

Figure 6. Target tracking results based on space coordinate system

In Figure 6, the dotted lines are real trajectories of targets, and the solid lines are the trajectories obtained by our method. As can be seen from the picture: near the center of the scene, we can get the space coordinates of the target accurately through the method we proposed; but at the edge of images, because of the camera distortion, the error will be relatively large. Overall, our approach can the trajectory of the target in space well.

5. CONCLUSION

Based on the analysis of the Zhengyou Zhang’s camera calibration method, we find that the relationship between the 2-D point coordinates and 3-D point coordinates is linear. Then we use deep learning method to obtain 2-D pixel coordinates corresponding to the spatial coordinates. In this paper, the spatial coordinates are obtained by manual measurement, and the GPS positioning technology can be used to obtain the spatial coordinates for the open scene. The conversion parameters between 2-D point coordinates and 3-D point coordinates can be solved through linear regression method. Finally, the target tracking in real space is realized based on the continuous position of the target in space. From the experimental results, it can be seen that: in the target detection stage, the deep learning technology can achieve high detection accuracy; and the coordinate transformation parameters obtained by the least squares method can well reflect the linear relationship between 2-D point coordinates and the 3-D point coordinates. So the track we finally get can be a good record of the target position change information.

ACKNOWLEDGEMENTS

This work was supported by the National Key Technologies Research and Development Program of China under Grant 2015BAD18B02.

REFERENCES

Abdel-Aziz, Y. I., Karara, H. M., and Hauck, M., 2015. Direct linear transformation from comparator coordinates into object space coordinates in close-range photogrammetry. *Photogrammetric Engineering & Remote Sensing*, 81(2), pp. 103-107.

Bourlard, H., Kamp, Y., 1988. Auto-association by multilayer perceptrons and singular value decomposition. *Biological Cybernetics*, 59(4-5), pp. 291-294.

Faig, W., 1975. Calibration of close-range photogrammetry systems: mathematical formulation. *Photogrammetric Engineering & Remote Sensing*, 41(12), pp. 1479-1486.

Girshick, R., Donahue, J., Darrell, T., and Malik, J., 2014. Rich feature hierarchies for accurate object detection and semantic segmentation. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 580-587.

Girshick, R., 2015. Fast R-CNN. In: *Proceedings of the 2015 IEEE International Conference on Computer Vision* IEEE Computer Society, pp. 1440-1448.

Hinton, G. E., Osindero, S., Teh, Y. W., 2006. A fast learning algorithm for deep belief nets. *Neural Computation*, 18(7), pp. 1527-1554.

Krizhevsky, A., Sutskever, I., and Hinton, G. E., 2012. ImageNet classification with deep convolutional neural networks. In: *International Conference on Neural Information Processing Systems*, Vol.60, pp. 1097-1105.

Luh, J. Y., Klaasen, J. A., 1985. A three-dimensional vision by off-shelf system with multi-cameras. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 7(1), pp. 35-45.

Lecun, Y., Bottou, L., Bengio, Y., and Haffner, P., 1998. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11), pp. 2278-2324.

Martins, H. A., Birk, J. R., and Kelley, R. B., 1981. Camera models based on data from two calibration planes. *Computer Graphics & Image Processing*, 17(2), pp. 173-180.

Migliore D. A., Matteucci M., Naccari M., 2006. A revaluation of frame difference in fast and robust motion detection. In: *Proceedings of the 4th ACM international workshop on Video surveillance and sensor networks* ACM, Vol.51, pp. 215-218.

Ren, S., He, K., Girshick, R., and Sun, J., 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. In: *Advances in neural information processing systems*, pp. 91-99.

Smolensky., 1986. *Information processing in dynamical systems: foundations of harmony theory*. MIT Press, Vol.1, pp. 194-281.

Tsai, R. Y., 1986. An efficient and accurate camera calibration technique for 3d machine vision. *Proc.ieee Conf.on Computer Vision & Pattern Recognition*, pp. 364-374.

Zhang, Z., 2000. A flexible new technique for camera calibration. *Tpami*, 22(11), pp. 1330-1334.

Zivkovic Z., 2004 Improved adaptive Gaussian mixture model for background subtraction. In: *Pattern Recognition, International Conference on IEEE Computer Society*, Vol. 2, pp. 28-31.

Zivkovic, Z., and Ferdinand, V. D. H., 2006. Efficient adaptive density estimation per image pixel for the task of background subtraction. *Pattern Recognition Letters*, 27(7), pp. 773-780.

Zhu, Q., Song, Z., Xie, Y., 2012. An efficient r-KDE model for the segmentation of dynamic scenes. In: *International Conference on Pattern Recognition* IEEE, pp. 198-201.