

A Smart City Assistive Infrastructure for the Blind and Visually Impaired People: A Thin Client Concept

Dmytro Zubov

University of Information Science and Technology "St. Paul the Apostle"

Partizanska, Ohrid, Macedonia (FYROM)

Phone: +389 46 511 000

dzubov@ieee.org

Abstract

The World Health Organization pointed out that over 285 million people worldwide suffer from loss of vision and blindness, and that the number could drop drastically in just a few years. About 90 % of the blind and visually impaired (B&VI) live at a low income that means these people cannot buy the expensive assistive devices for the spatial cognition. In this work, the new concept of the smart city assistive infrastructure with distributed server-client architecture is presented for the B&VI using the inclusive smart assistive component that interacts with other subsystems of the smart city (smart buildings, smart mobility, smart energy, etc.) via IoT protocols such as MQTT. The main constituents of thin client are as follows: Raspberry Pi 3 B board with camera for the objects detection / recognition, ultrasonic sensor(s) HC-SR04 for the obstacles identification on the short-range distance up to 5 m, GPS module for the global navigation, iBeacon Bluetooth low energy proximity sensing software, MQTT IoT protocol for the mutual communication of clients, Python multithread application, Raspbian OS. The thin client hardware is of affordable price USD 70. The objects detection and recognition are implemented on the thin clients via the Histogram of Oriented Gradients with Euclidean distance classifier (HOG+EDC). The design of the recognition models, the file hosting of the training images and the knowledge base with the recognition rules are done on server(s). The modified Viola-Jones fast face detector with the combination of features "eye" and "nose" is proposed to speed up the image processing, but its detection rate is not 100 %. Hence, it can be applied only with the subsequent recognition using HOG+EDC.

Keywords: Smart City, Assistive Device, B&Vi People, Hog, Raspberry Pi 3 B.

1. Introduction

The World Health Organization pointed out that over 285 million people are estimated to be blind or visually impaired (B&VI) worldwide nowadays. About 90 % of the world's visually impaired live at low income, that means these people are not able to buy the expensive assistive devices for the spatial cognition. These devices play an important role in the outdoor exploration of the B&VI, their social integration and independent travel including the school, college, and university studies. Approximately 20 % of young B&VI in the UK do not leave their home, approximately 30 % had traveled locally, and only 40 % left their home alone and walked (Bruce et al., 1991). Most of the B&VI who explore new routes feel disorientation, fear, stress, and panic associated with being lost (Golledge, 1993). Being mobile is the crucial factor that contributes to the success of the B&VI (Goodwyn, Bell, & Singletary, 2009). The major problems are as follows:

1. The high price of existing assistive devices for B&VI, e.g. USD 10,000 Brainport V100 (BrainPort V100 Device, 2015) and Euro 2,000 EVA (EVA Report Summary, 2018).

2. The high computational complexity of existing image processing algorithms, e.g. Viola-Jones and Histogram of Oriented Gradients (HOG) with Support Vector Machine (SVM) (Xu et al., 2016), on the one hand and the low performance of the wearable Internet of Things (IoT) devices, e.g. LilyPad Arduino and Raspberry Pi 3 B (Patnaik Patnaikuni & Dinkar, 2017), on the other hand. In this case, the cloud computing resources, e.g. Amazon AWS Rekognition (Amazon Rekognition, 2018), might be used for the objects detection and recognition, where low-performance devices work as the thin clients. However, this approach requires a fast and stable Internet connection and

payment for the number of images and/or minutes of video analyzed and/or stored in the computational cloud, which is impossible to realize widely nowadays due to unstable Internet connection and limited budget of social projects. In addition, the software with high computational complexity algorithms and/or permanent Internet connection consume more energy than the average.

3. Lack of the funds for social projects including the development of new assistive devices for the B&VI, production or assembling of components for assistive devices using assembly-ready components (e.g. Raspberry Pi and Arduino boards and sensors), as well as the after-sale support and/or maintenance.

4. The infrastructure of existing smart cities does not have unified soft- and hardware interfaces for the integration of assistive devices for the B&VI.

5. Guide dogs offer huge support, but with the cost of keeping and training, the dog approaching Euro 60,000. It is not available to most B&VI (EVA Report Summary, 2018).

The main goal and motivation of this work is the development of the wearable assistive device for the B&VI with affordable price less than USD 100 and fast image processing algorithm. A distributed server-client architecture (Ride et al., 2012) is proposed for the soft- and hardware complex. The computationally optimized Viola-Jones method and HOG with Euclidean distance classifier (EDC) are suggested to be the basic methods for the real-time detection and recognition. The face detection and recognition is a dominant problem for the spatial cognition of B&VI, but other objects, e.g. animals and cars, might be detected and recognized as well. The single-board computer Raspberry Pi 3 B with the camera for the image processing, iBeacon Bluetooth low energy proximity sensing software (Trent et al., 2017), and an ultrasonic sensor for the distance measurement to the obstacles are proposed to be the basic hardware for the thin client due to the low price (up to USD 100), small size (average hand), lightweight (approx. 0.2 kg), 5 V power supply (mobile power banks also can be used), and onboard network devices (802.11 b/g/n Wireless LAN, 10/100 Ethernet adapter, Bluetooth 4.1). Other compatible sensor modules can be installed on the board if necessary. The recognition models are designed on the server side due to the high computational complexity and need expert knowledge. The images are processed on the thin clients.

This paper is organized as follows: In Section 2, the related work and previous studies are presented. In Section 3, a case study of Viola-Jones fast face detector with the combination of features “eye” and “nose” is discussed. In Section 4, the objects detection and recognition with the histogram of oriented gradients and Euclidean distance classifier are presented. In Section 5, a smart city assistive infrastructure for the B&VI people is suggested. Conclusions are summarized in Section 6.

2. Related work

In (Zubov, 2017), the ubiquitous assistive device was developed for the spatial cognition of B&VI by measuring the distance to the obstacle using ultrasonic sensor HC-SR04, Arduino Uno, MP3 player with SD card, and headphone. The distance is pronounced to the B&VI via headphone. The devices of the budget price USD 10 each were successfully tested by the B&VI at the Instituto para Ciegos y Débiles Visuales “Ezequiel Hernández Romo” (San Luis Potosi, Mexico) and Lions Clubs International (Mexico, Distrito B3). Another presented device supports the golf game of B&VI. Every golf flagstick has the sound marking based on the Wi-Fi board NodeMcu Lua ESP8266 ESP-12 with an active buzzer. The Wi-Fi boards are controlled remotely by the person with good vision via Intranet HTML websites. Mini portable Wi-Fi router securely connects all devices in a local area network. Ten assistive devices of the budget price USD 7 each were successfully tested and handed to Instituto para Ciegos y Débiles Visuales “Ezequiel Hernández Romo” together with Wi-Fi router.

In (Elmannai & Elleithy, 2017), twenty five different assistive devices are compared using five essential characteristics: real-time / not real-time, coverage (indoor, outdoor, both), time (day, night, both), range (less than 1 m, 1 m to 5 m, greater than 5 m), object type (static, dynamic, both).

72 % of the presented products have at least three characteristics that are not fully satisfied. In particular, device Eye Subs provides outdoor coverage but not indoor coverage, the detection range is less than 5 m due to the ultrasonic limitation, and it detects only the static but not the dynamic objects. None of the presented systems is 100 % satisfactorily in terms of the essential characteristics. In addition, the prices are not shown. (Elmannai & Elleithy, 2017) pointed out that the ideal device has to satisfy the user needs, i.e. to recognize the surrounding environment at all times and everywhere. The system cannot be limited by a specific case since the soft- and hardware complex has to be modifiable.

In (Poggi & Mattoccia, 2016), the wearable mobility aid device is presented with custom RGBD camera and an embedded FPGA CPU. The semantic categorization of detected obstacles is based on the convolutional neural network. The experimental results showed good detection performance (close to 98 % of detection rate), although the object categorization is less than 72 % of correctness. The sensed area for obstacle detection is set from 0.5 m to 3 m only, and the price is not shown.

In (Jacobson & Kitchin, 1997), the GIS manages inputs from obstacle avoidance sensors, positioning devices, and the end-user, referencing them to complex spatial databases. Then, this spatial information is conveyed to the end-user via audio- and / or tactile-based interfaces forward for few steps. Nowadays, the GPS includes GIS, and hence the B&VI people have explicit navigation instructions.

In (Ramadhan, 2018), the wearable smart navigation system is presented using GPS and ultrasonic sensor. A solar panel provides the additional power to the system. However, the image processing component is absent.

(Gori et al., 2016) showed that the technological devices specifically designed for the B&VI are not easily adaptable to the children since they require too much attention and long training periods for the normal exploitation of devices. In 2014, the number of visually impaired children below the age of 15 was estimated to 19 million worldwide, and hence this problem needs special attention. The proposed solution consists in the additional training for children.

In (Bai et al., 2017), the depth-based way-finding algorithm finds the candidate traversable directions based on the depth image. Different from the floor-segmentation based way-finding methods, it only uses the region of interest to determine the traversable directions. It is assumed that the nearest obstacle is always at the bottom of the depth image, and it selects only a line in the bottom of the image as input. The ultrasonic sensor is used as well. In this work, the object recognition is not discussed.

(Trent et al., 2017) stated that none of the existing navigation assistive systems for blind people is integrated using IoT techniques. The low cost and low power IoT navigation system for blind people is proposed. The Raspberry Pi board detects the obstacles with the ultrasonic sensor. The information is provided to the end-user via a Bluetooth headset. iBeacon technology is proposed for the spatial marking.

Analysis of the above-stated work shows that the spatial cognition is based on the ultrasonic measurement of the distances to obstacles, GPS navigation, and the detection and recognition of the objects in the surrounding environment using an image processing implemented on high-performance hardware. None of the known inclusive smart assistive subsystems has the distributed server-client architecture, as well as the integration of the assistive devices into the smart city infrastructure has not been proposed.

Nowadays, the distances to obstacles are measured by different well-developed sensors like ultrasonic HC-SR04 (Elmannai & Elleithy, 2017) and optical infrared Sharp GP2Y0A21YK0F (Adarsh et al., 2016; Masuda et al., 2001). The GPS navigation is a standard subsystem in up-to-date smartphones that provides explicit instructions for the route(s) establishment. However, the objects detection and recognition methods are well developed for the high-performance hardware only, the low-performance boards, e.g. Raspberry Pi 3 B and Arduino Uno, require modified and / or computationally optimized methods to provide the real-time information to the end-users. There are

different types of image processing algorithms (Mallick, 2016; Salton, 2017; Takuya et al., 2008; Viola & Jones, 2004), for example:

- Eigenfaces;
- Local Binary Patterns Histograms;
- Fisherfaces;
- Scale Invariant Feature Transform;
- Speed Up Robust Features;
- Viola-Jones Face Detection Algorithm;
- HOG.

These methods have different approaches to extract the image features and perform the matching with the input image. Some methods, like HOG, can be used for the detection and recognition of different classes of the objects, e.g. cars and animals. The SVM classifier with the HOG feature extractor is the most popular solution for object detection and classification (Ilas & Ilas, 2018). (Marco et al., 1987) found that the sample linear discriminant function (LDF) performs poorly when the number of features is large, relative to the size of the training samples. In particular, a comparison of the LDF and EDC based on the classification of the individuals from two spherical normal populations showed that the sample EDC outperforms the sample LDF when the number of features is large relative to the training sample size. In general, the selection of the method depends on the specificity of the problem. In this paper, Viola-Jones fast face detection and HOG algorithms are discussed in detail. The initial idea of this work was to detect the face via the fast Viola-Jones algorithm with the proposed combination of features “eye” and “nose”, and then apply HOG method for the face recognition. As it is shown below, the Viola-Jones detector is 100 % accurate on the training and validation samples but cannot detect correctly faces on the test sample. At the same time, the HOG+EDC method shows 100 % accuracy on the training, validation, and test samples.

3. Viola-Jones real-time face detector: a case study on the combination of features “eye” and “nose”

The Viola-Jones fast face detector is based on the AdaBoost learning algorithm with the given feature set and training sample of positive and negative images. The integral image is calculated at the location (x, y) as the sum of the pixel intensities above and to the left of the point with coordinates (x, y) inclusive (Viola & Jones, 2004):

$$ii(x, y) = \sum_{x' \leq x, y' \leq y} i(x', y'), \quad (1)$$

where $i(x, y)$ is the pixel value of the original image, $ii(x, y)$ is the corresponding image integral value. The sum of the pixel intensities on any rectangular area $ABCD$ is computed with only four values $ii(A)$, $ii(B)$, $ii(C)$, and $ii(D)$ extracted from the integral image:

$$\sum_{(x, y) \in ABCD} i(x, y) = ii(D) + ii(A) - ii(B) - ii(C). \quad (2)$$

Different Haar-like features are used in the enclosing detection window. The two-rectangle horizontal and three-rectangle vertical features are usually associated with the eye(s) and nose representation on the face. In this work, the two-rectangle vertical feature located just below the two-rectangle horizontal one is used as the nose representation on the face. A Haar-like feature value f is a difference between the pixel sums of two rectangles.

The Viola-Jones method uses a variant of AdaBoost classifier to select a small set of Haar-like features and train the classifier. A single AdaBoost classifier consists of a weighted sum of many weak classifiers, where each weak classifier is a threshold on a single Haar-like rectangular feature:

$$h(w_s, f, p, T) = \begin{cases} 1 & pf(w_s) < pT \\ 0 & \text{otherwise} \end{cases}, \quad (3)$$

where T is the threshold, p is the polarity indicating the direction of the inequality, w_s is a subwindow of an image.

In this work, a new feature is proposed as the combination of horizontal and vertical two-rectangle features (see Figure 1):

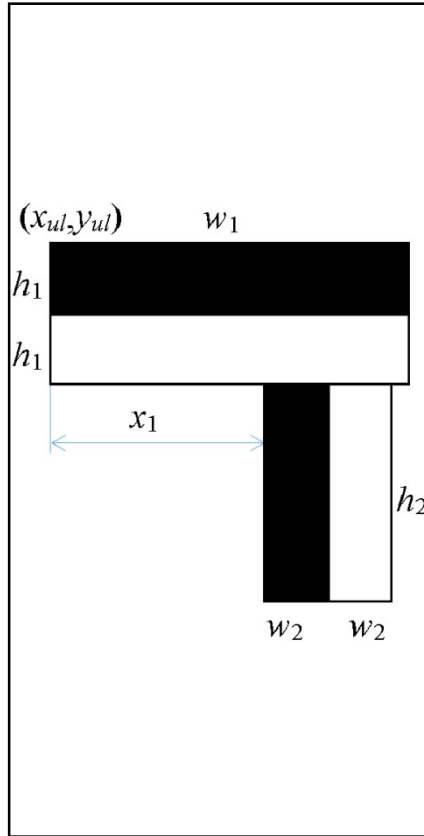


Figure 1. Combination of horizontal and vertical two-rectangle features shown in the enclosing detection window.

In Figure 1, (x_{ul}, y_{ul}) is a relative coordinate of the upper-left pixel of the horizontal two-rectangle feature in the enclosing detection window of size 64x128, x_1 is a horizontal shift right of the vertical feature according to the horizontal one, h_1 (w_1) and h_2 (w_2) are the height (width) of the horizontal and vertical two-rectangle features, respectively. The vertical two-rectangle feature is located just below the horizontal one. The following inequalities describe the possible sizes and locations of these features in the enclosing detection window:

$$(x_1 + w_2 + w_2) \leq w_1, \quad (4)$$

$$w_1 \leq 64, \quad (5)$$

$$(2h_1 + h_2) \leq 128, \quad (6)$$

The optimization task is to identify the positions and dimensions of horizontal and vertical two-rectangle features. The criterion is the minimization of the wrong detections on the validation sample. The training and test samples contain one image each, the validation sample – 60 images (see Figure 2):



Figure 2. The images of the training (A), validation (B), and test (C) samples.

The images of the training and validation samples are of 64x128 size. The combinatorial optimization has a high computational complexity, and hence the following values of h_1 and w_1 are proposed to reduce the number of operations on the Viola-Jones detector:

- 1st group with ratio 1/3: 1/3, 2/6, 3/9, 4/12, 5/15, 6/18, 7/21, 8/24, 9/27, 10/30, 11/33, 12/36, 13/39, 14/42, 15/45, 16/48, 17/51, 18/54, 19/57, 20/60;
- 2nd group with ratio 1/4: 1/4, 2/8, 3/12, 4/16, 5/20, 6/24, 7/28, 8/32, 9/36, 10/40, 11/44, 12/48, 13/52, 14/56, 15/60;
- 3rd group with ratio 1/5: 1/5, 2/10, 3/15, 4/20, 5/25, 6/30, 7/35, 8/40, 9/45, 10/50, 11/55, 12/60;
- 4th group with ratio 1/6: 1/6, 2/12, 3/18, 4/24, 5/30, 6/36, 7/42, 8/48, 9/54, 10/60.

For instance, the set of possible values at $h_1=2$ and $w_1=6$ pixels is as follows:

$$\left\{ \begin{array}{l} (h_1=2, w_1=6, h_2=3, w_2=1, x_1=0), (h_1=2, w_1=6, h_2=3, w_2=1, x_1=1), \\ (h_1=2, w_1=6, h_2=3, w_2=1, x_1=2), (h_1=2, w_1=6, h_2=4, w_2=1, x_1=0), \\ (h_1=2, w_1=6, h_2=4, w_2=1, x_1=1), (h_1=2, w_1=6, h_2=4, w_2=1, x_1=2), \\ (h_1=2, w_1=6, h_2=5, w_2=1, x_1=0), (h_1=2, w_1=6, h_2=5, w_2=1, x_1=1), \\ (h_1=2, w_1=6, h_2=5, w_2=1, x_1=2), (h_1=2, w_1=6, h_2=6, w_2=1, x_1=0), \\ (h_1=2, w_1=6, h_2=6, w_2=1, x_1=1), (h_1=2, w_1=6, h_2=6, w_2=1, x_1=2), \\ (h_1=2, w_1=6, h_2=6, w_2=2, x_1=0), (h_1=2, w_1=6, h_2=8, w_2=2, x_1=0), \\ (h_1=2, w_1=6, h_2=10, w_2=2, x_1=0), (h_1=2, w_1=6, h_2=12, w_2=2, x_1=0) \end{array} \right\}$$

The total number of possible combinations $(h_1, w_1, h_2, w_2, x_1)$ is only 57,082 using the four proposed groups.

The software was developed in Python programming language and can be executed directly on the Raspbian operating system (OS) (Pajankar, 2017) and / or on Windows 8 / 10 OS. In the last case, Anaconda distribution of the Python programming language with additional packages (Anaconda Cloud, 2018) is installed additionally.

Analysis of the faces detection and recognition on the training and validation samples shows that the presented combination of horizontal and vertical two-rectangle features generates several recognition models without mistakes. An example: $h_1=8$; $w_1=24$; $h_2=60$; $w_2=8$; $x_1=0$; values of the horizontal and vertical features must be less than -20793 and -21386, respectively. The test image is of 378x504 size (see Figure 2, C). The enclosing detection window was slid in the test images of the scale sizes 64x85, 80x106, 100x133, 125x166, 156x208, 195x260, 243x324, 303x404, and 378x504, i.e. with ratio 1.25. Analysis of the results shows that all generated recognition models detected the training image on the test image with resolution 243x324 at the point (175, 46) that is

wrong. Hence, the Viola-Jones fast face detector can be considered only as a preliminary stage for the subsequent face recognition based on a more accurate method such as HOG.

4. Objects detection and recognition with the histogram of oriented gradients and Euclidean distance classifier

(Pedersoli et al., 2007) showed that Haar-like features have little discriminative power, but HOG gives excellent results in human detection and recognition, as well as HOG features can be calculated in a constant time. The input image is of size 64x128. For the HOG features extraction, the horizontal and vertical convolutions with $[-1 \ 0 \ 1]$ and $[-1 \ 0 \ 1]^T$ masks are applied to every pixel, where $[\]^T$ is a transpose operation. (Takuya et al., 2008) define explicitly x -directional (g_x) and y -directional (g_y) gradients. If a pixel is absent in the convolution operation due to the border of the image, it is considered 0. Then, the gradient magnitude $g(x,y)$ and orientation $\theta(x,y)$ are calculated using the gradients g_x and g_y as follows:

$$g(x, y) = \sqrt{g_x^2 + g_y^2}, \quad (7)$$

$$\theta^*(x, y) = \arctan \frac{g_y}{g_x}, \quad (8)$$

$$\theta(x, y) = \begin{cases} \theta^*(x, y) & \text{if } \theta^*(x, y) < \pi \\ \theta^*(x, y) - \pi & \text{if } \theta^*(x, y) > \pi \end{cases}. \quad (9)$$

The standard trigonometric function `math.atan2` of the Python programming language implements the formula (8).

The local region 64x128 is divided into small cells 8x8. The nine features are an output of every cell (Mallick, 2016) – these are 9 bins corresponding to angles 0, 20, 40, ..., 160. Then, the 16x16 block normalization is applied with L^2 norm of the vector. There are seven horizontal and fifteen vertical possible positions of 16x16 blocks making a total of $7*15=105$ possible positions. Each 16x16 block is represented by 36x1 vector. When they are concatenated into one feature vector, the $36*105=3780$ dimensional vector is obtained. This vector contains 3780 HOG features f_H .

The EDC uses the following formula (Salton, 2017):

$$D = \sqrt{\sum_{i=1}^{3780} (f_{H1,i} - f_{H2,i})^2}, \quad (10)$$

where $f_{H1,i}$ and $f_{H2,i}$ are the features of the training and validation / test images, respectively. In this work, the test / validation image is considered as an instance of the class of the training images if D is less than threshold T .

The following set of the training images is proposed with the various possible positions of the camera and the training face(s) (see Figure 3; the images are numbered from left to right and from top to bottom):

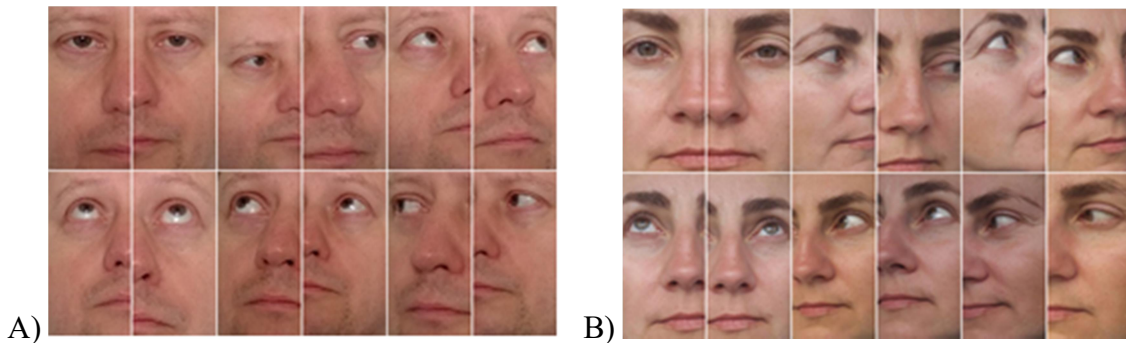



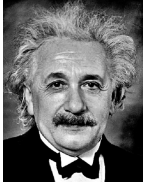








Figure 3. Two sets of training images.

In Figure 3, the positions of face on the training images are as follows: straight, left, up / left, upward, up / right, right. The positions down / right, downward, down / left are not included due to closed eyes, i.e. they are not considered informative. The training images are of size 64x128, and hence two areas are located on the faces, i.e. every position of the face is represented by two training images. Results of ten calculations of Euclidean distances, relative coordinates of upper-left pixels of enclosing detection windows 64x128, sizes of the scaled images, and number of the training image in Figure 3, A are shown in table 1. The input images are scaled with ratio 1.25.

Table 1. Calculation of Euclidean distance for the set of training images presented in Figure 3, A

#	Test image and its size	Euclidean distance / relative coordinate of the upper-left pixel / size of the image / number of the training image in Figure 3, A	Instance of the training or other class	#	Test image and its size	Euclidean distance / relative coordinate of the upper-left pixel / size of the image / number of the training image in Figure 3, A	Instance of the training or other class
1	 582x750	5.01 / (188, 125) / 378x487 / 2	Training class	6	 350x481	6.38 / (143, 120) / 303x416 / 2	Other class
2	 475x656	6.01 / (18, 61) / 156x215 / 3	Training class	7	 476x670	6.11 / (230, 113) / 472x664 / 2	Other class
3	 471x569	6.24 / (27, 49) / 195x235 / 3	Other class	8	 478x512	5.89 / (50, 14) / 472x505 / 2	Other class
4	 472x629	6.72 / (343, 325) / 472x629 / 1	Other class	9	 375x555	6.32 / (76, 211) / 243x359 / 1	Other class
5	 472x629	5.07 / (110, 112) / 303x404 / 1	Training class	10	 423x688	6.27 / (40, 104) / 195x317 / 7	Other class

Analysis of the Euclidean distances presented in table 1 shows that the images of training class are detected and recognized correctly if $D < 5.1$, i.e. $T = 5.1$. This classification rule describes the algorithm of EDC. The images are classified into two groups (belong or not belong to the training

class) based on the classification rule, and hence the presented algorithm of EDC is binary. The inequality $D > T$ for the second image is explained by the fact that the photo was taken nine years ago using a different camera and its features are different to the latest one. The similar results were achieved for the training images from Figure 3, B. The images 3 and 9 in table 1 are instances of the class of training images in Figure 3, B.

The selection and preprocessing of the training images is a high-qualified work done by the expert, and hence it is proposed to host them together with the recognition knowledge base on the Internet file hosting. They are downloaded when the assistive device is started. If there is no Internet connection, the training images and the knowledge base with the recognition rules are loaded from the local storage. In a future research, the HOG algorithm will be optimized for the soft- and hardware. Also, SVM (Takuya, 2008) can replace ECD if the classification rules do not detect and recognize objects correctly.

5. A smart city assistive infrastructure for the blind and visually impaired people

There is no standardized and commonly accepted definition for the smart city (Smart cities and infrastructure, 2016; Al-Hader & Rodzi, 2009). A concept of the inclusive smart infrastructure (de Oliveira Neto & Kofuji, 2016), where physical and digital barriers are eliminated, targets the needs of the B&VI at most. The best implementation of this concept is based on the data-driven approach, which uses the IoT protocols such as MQTT, CoAP, XMPP, DDS, AMQP (Dizdarevic et al., 2018). Figure 4 demonstrates the example of the thin clients' interaction via MQTT IoT protocol that takes approximately 10 KB of the random access memory:

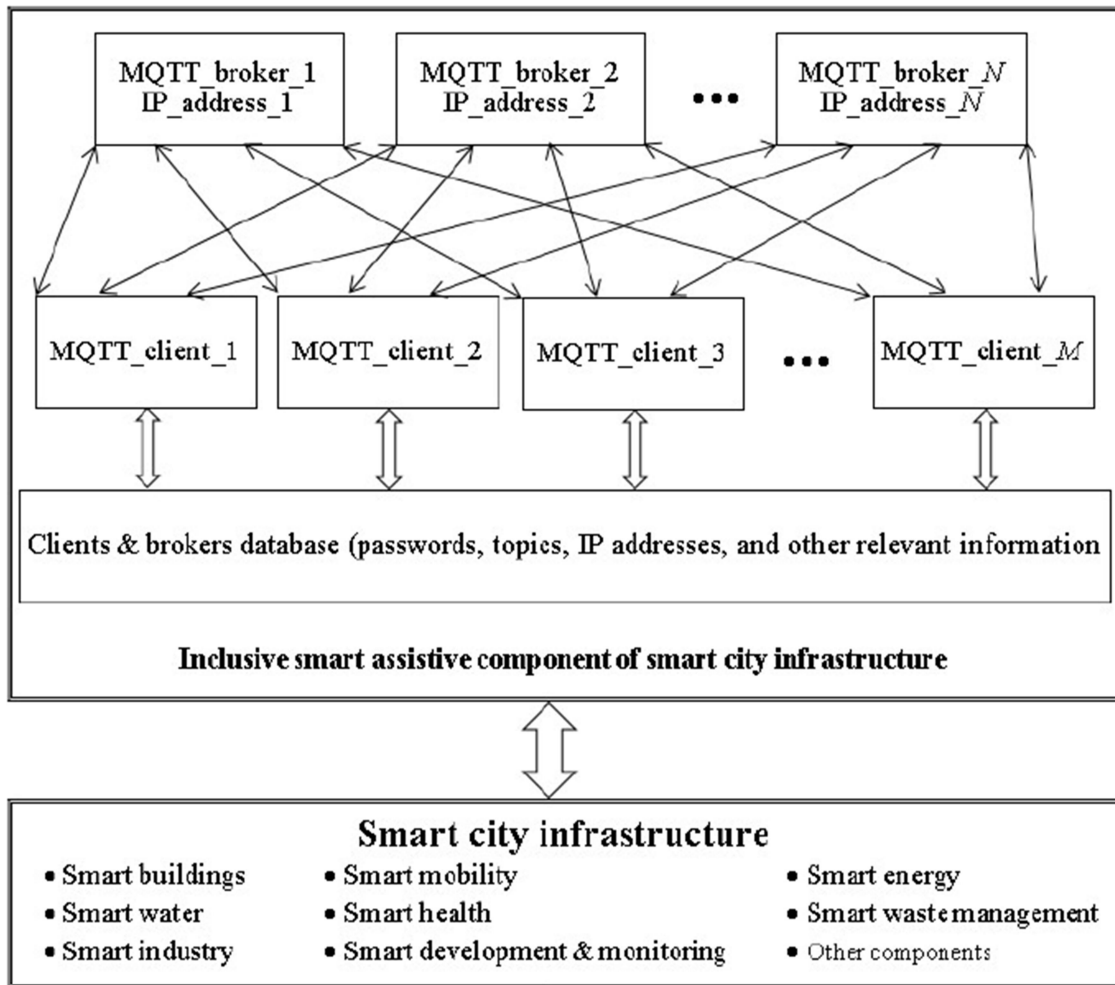


Figure 4. Example of the smart city data-driven assistive infrastructure for the B&VI people using MQTT IoT protocol.

The presented data-driven assistive infrastructure is centralized around the broker(s) on the one hand and decentralized on the other hand due to the mutual communication of clients via different brokers. An assistive smart constituent of a smart city interacts with other components (smart buildings, smart mobility, smart energy, smart water, smart health, smart waste management, smart industry, smart infrastructure development and monitoring, etc.) via IoT protocols, WebAPI, storage, and other communication technologies. Every MQTT client optionally includes the following navigation hardware for the spatial cognition:

- GPS module;
- iBeacon Bluetooth low energy proximity sensing software (Raboy, 2016);
- ultrasonic sensor(s) for the obstacles identification on the short-range distance up to 5 m;
- camera for the objects detection and recognition.

The Python programming language and the Raspberry Pi 3 B board are considered as a basic soft- and hardware in this project. Raspberry Pi 3 B board with Pi camera and ultrasonic sensor HC-SR04 are shown in Figure 5:

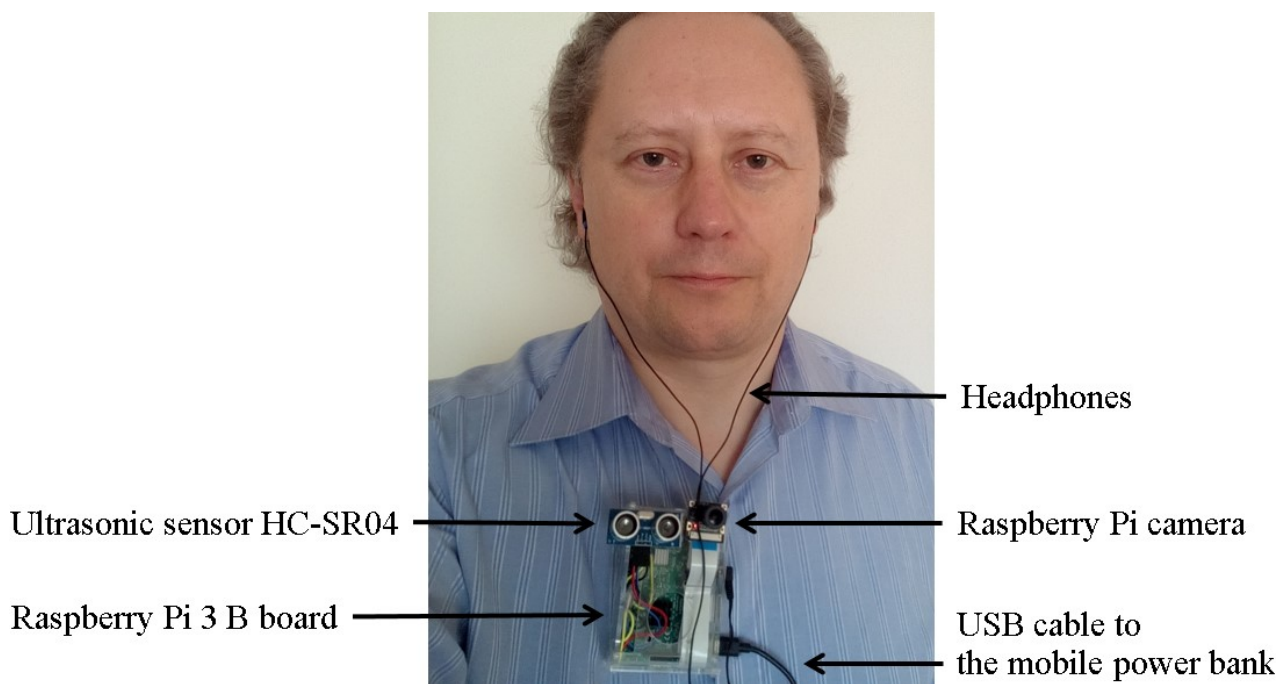


Figure 5. Raspberry Pi 3 B board with Pi camera and ultrasonic sensor HC-SR04.

The HC-SR04 ultrasonic range sensor is installed with two resistors 1 k Ω and 2 k Ω (HC-SR04 Ultrasonic Range Sensor, 2014). The Raspberry Pi 3 B uses a quad-core processor ARM Cortex-A53, and hence the multithread application is proposed to implement the above-stated spatial cognition functions. At least one thread executes the objects detection and recognition due to the high computational complexity of the image processing algorithms.

The price of the thin client hardware is USD 70 approx.

6. Conclusions

In this paper, a new smart city assistive infrastructure for the B&VI is presented. The assistive cyber-physical component interacts with other subsystems (smart buildings, smart mobility, smart energy, smart water, smart health, smart waste management, smart industry, smart infrastructure development and monitoring, etc.) via up-to-date communication technologies such as MQTT IoT protocol. A distributed server-client architecture is suggested for the soft- and hardware complex. The thin clients with affordable price USD 70 each consist of the following:

- Raspberry Pi 3 B board with a camera for the objects detection / recognition, ultrasonic sensor(s) HC-SR04 for the obstacles identification on the short-range distance up to 5 m, and GPS module for the global navigation;
- iBeacon Bluetooth low energy proximity sensing software;
- MQTT IoT protocol for the communication of MQTT clients;
- Python multithread application;
- Raspbian OS.

The objects are detected and recognized on the thin clients via the HOG+EDC method. The design of recognition models is implemented on the server. The training images and the knowledge base with recognition rules are hosted on the Internet file hosting service. They are downloaded when the assistive device is started. If there is no Internet connection, the training images and the knowledge base with recognition rules are loaded from the local storage.

The modified Viola-Jones fast face detector with the combination of features “eye” and “nose” is proposed to speed up the image processing. The detection rate is not 100 %, and hence it can be applied only with the subsequent recognition using HOG+EDC(SVM).

The most likely prospect of the presented smart infrastructure is the development of the assistive environment for a group of B&VI people.

References

- Adarsh, S., Kaleemuddin, S.M., Bose, D., & Ramachandran, K.I. (2016). Performance comparison of Infrared and Ultrasonic sensors for obstacles of different materials in vehicle/ robot navigation applications. *IOP Conference Series: Materials Science and Engineering*, Vol. 149, Iss. 1. Retrieved from <http://iopscience.iop.org/article/10.1088/1757-899X/149/1/012141>
- Al-Hader, M., & Rodzi, A. (2009). The smart city infrastructure development & monitoring. *Theoretical and Empirical Researches in Urban Management*, 4 (2 (11)), 87-94. Retrieved from https://www.researchgate.net/publication/46567764_The_smart_city_infrastructure_development_monitoring
- Anaconda Cloud. (2018). Retrieved from <https://anaconda.org>
- Amazon Rekognition. Deep learning-based image and video analysis. (2018). Retrieved from <https://aws.amazon.com/rekognition/>
- Bai, J., Lian, S., Liu, Z., Wang, K., & Liu, D. (2017). Smart guiding glasses for visually impaired people in indoor environment. In: *IEEE Transactions on Consumer Electronics*, Vol. 63, no. 3, 258-266, August 2017.
- BrainPort V100 Device Helps People who are Blind See with Tongue. (2015). Retrieved from <https://brailleworks.com/brainport-v100/>
- Bruce, I., McKennell, A., & Walker, E. (1991). *Blind and Partially Sighted Adults in Britain: The RNIB Survey*. Vol. 1, London, Her Majesty's Stationery Office.
- Dizdarevic, J., Carpio, F., Jukan, A., & Masip-Bruin, X. (2018). Survey of Communication Protocols for Internet-of-Things and Related Challenges of Fog and Cloud Computing Integration. Retrieved from <https://arxiv.org/abs/1804.01747>
- de Oliveira Neto, J.S., & Kofuji, S.T. (2016). Inclusive Smart City: An Exploratory Study. In: Antona, M., Stephanidis, C. (eds) *Universal Access in Human-Computer Interaction. Interaction Techniques and Environments*. UAHCI 2016. Lecture Notes in Computer Science, Vol. 9738. Springer, Cham.
- Elmannai, W. & Elleithy, K. (2017). Sensor-Based Assistive Devices for Visually-Impaired People: Current Status, Challenges, and Future Directions. *Sensors (Basel)*, 2017 Mar 10, 17(3).
- EVA Report Summary. (2018). Retrieved from https://cordis.europa.eu/result/rcn/223928_en.html
- Golledge, R.G. (1993). Geography and the Disabled: A Survey with Special Reference to Vision Impaired and Blind Populations. *Transactions of the Institute of British Geographers*, 18(1), 63-85.

- Goodwyn, M., Bell, E.C., & Singletary, C. (2009). Factors that Contribute to the Success of Blind Adults. Research Report of the Professional Development and Research Institute on Blindness. Ruston: Louisiana Tech University. Retrieved from <http://www.pdrib.com/downloads/Factors%20that%20Contribute%20to%20the%20Success%20of%20Blind%20Adults.doc>
- Gori, M., Cappagli, G., Tonelli, A., Baud-Bovy, G., & Finocchietti, S. (2016). Devices for visually impaired people: High technological devices with low user acceptance and no adaptability for children. *Neuroscience & Biobehavioral Reviews*, Vol. 69, 79-88.
- HC-SR04 Ultrasonic Range Sensor on the Raspberry Pi. (2014). Retrieved from <https://www.modmypi.com/blog/hc-sr04-ultrasonic-range-sensor-on-the-raspberry-pi>
- Ilas, M.-E., & Ilas, C. (2018). A New Method of Histogram Computation for Efficient Implementation of the HOG Algorithm. Retrieved from www.mdpi.com/2073-431X/7/1/18/pdf
- Jacobson, R.D. & Kitchin, R.M. (1997). GIS and people with visual impairments or blindness: Exploring the potential for education, orientation, and navigation. *Transactions in GIS*, 2(4). 315-332.
- Mallick, S. (2016). Histogram of Oriented Gradients. Retrieved from <https://www.learnopencv.com/histogram-of-oriented-gradients/>
- Marco, V.R., Young, D.M., & Turner, D.W. (1987). The Euclidean Distance Classifier: An Alternative to the Linear Discriminant Function. *Communications in Statistics – Computation and Simulation*, 16, 485-505.
- Masuda, R., Sasa, S., & Hasegawa, K. (2001). Optical Proximity Sensor by using Phase Information *Trans. of the Society of Instrument and Control Engineers*, Vol. E-1, no.1, 1/7, 6-12. Retrieved from <http://sice.sakura.ne.jp/e-trans/papers/E1-2.pdf>
- Pajankar, A. (2017). Raspberry Pi Image Processing Programming: Develop Real-Life Examples with Python, Pillow, and SciPy. New York: Apress.
- Patnaik Patnaikuni & Dinkar R. (2017). A Comparative Study of Arduino, Raspberry Pi and ESP8266 as IoT Development Board. *International Journal of Advanced Research in Computer Science*. May/Jun2017, Vol. 8, Iss. 5, 2350-2352. Retrieved from <http://www.ijarcs.info/index.php/Ijarcs/article/download/3959/3721>
- Pedersoli, M., González, J., Chakraborty, B., & Villanueva, J.J. (2007). Enhancing Real-Time Human Detection Based on Histograms of Oriented Gradients. In: Kurzynski, M., Puchala, E., Wozniak, M., Zolnierok, A. (eds) *Computer Recognition Systems 2. Advances in Soft Computing*, Vol. 45. Springer, Berlin, Heidelberg.
- Poggi, M. & Mattoccia, S. (2016). A Wearable Mobility Aid for the Visually Impaired based on embedded 3D Vision and Deep Learning. 2016 *IEEE Symposium on Computers and Communication (ISCC)*, Messina, 208-213.
- Raboy, N. (2016). Scan For Bluetooth Enabled iBeacons Via A Raspberry Pi IoT Device. Retrieved from <https://www.thepolyglotdeveloper.com/2016/09/scan-bluetooth-enabled-ibeacons-via-raspberry-pi-iot-device/>
- Ramadhan, A.J. (2018). Wearable Smart System for Visually Impaired People. *Sensors*, 18, 843. Retrieved from <http://www.mdpi.com/1424-8220/18/3/843/htm>
- Ride, J.R., James, D.A., Lee, J.B., & Rowlands, D.D. (2012). A distributed architecture for storing and processing multichannel multi-sensor athlete performance data. *Procedia Engineering*, 34, 403-408. Retrieved from https://research-repository.griffith.edu.au/bitstream/handle/10072/47080/78905_1.pdf?sequence=1
- Salton, K. (2017). Face Recognition: Understanding LBPH Algorithm. Retrieved from <https://towardsdatascience.com/face-recognition-how-lbph-works-90ec258c3d6b>

- Smart cities and infrastructure. Commission on Science and Technology for Development. Nineteenth session (2016). Retrieved from http://unctad.org/meetings/en/SessionalDocuments/ecn162016d2_en.pdf
- Takuya, K., Akinori, H., & Takio K. (2008). Selection of Histogram of Oriented Gradients Features for Pedestrian Detection. In: Ishikawa, M. et al. (Eds.): ICONIP 2007, Part II, LNCS 4985, 598-607.
- Trent, M., Abdelgawad, A., & Yelamarthi, K. (2017). A Smart Wearable Navigation System for Visually Impaired. In: Gaggi, O., Manzoni, P., Palazzi, C., Bujari, A., & Marquez-Barja, J. (eds) Smart Objects and Technologies for Social Good. GOODTECHS 2016. Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, Vol. 195. Springer, Cham.
- Viola, P. & Jones, M.J. (2004). Robust Real-Time Face Detection. *International Journal of Computer Vision*, 57(2), 137-154.
- Xu, Y., Yu, G., Wang, Y., Wu, X., & Ma, Y. (2016). A Hybrid Vehicle Detection Method Based on Viola-Jones and HOG+SVM from UAV Images. *Sensors* (Basel, Switzerland), 16(8), 1325. Retrieved from <http://researcher.watson.ibm.com/researcher/files/us-bennetc/BB84highest.pdf>
- Zubov, D. (2017). *A Case Study on the Spatial Cognition of Surrounding Objects by the B&VI People Using Sound Patterns and Ultrasonic Sensing*. In Kocovic, P., Behringer, R., Ramachandran, M., Mihajlovic, R. (Eds.), *Emerging Trends and Applications of the Internet of Things*, 105-116. Hershey, USA: IGI Global.



Dmytro Zubov (b. Dec. 8, 1972) received Dipl. Engineering at V.Dahl East Ukrainian National University in 1995, PhD in Engineering (lower-level doctorate) at Donetsk National Technical University in 1998, and then PhD in Engineering (higher-level doctorate) at Kryvyi Rih Technical University in 2006. He was visiting professor in Computer Science at Tecnológico de Monterrey, Universidad Politécnica de San Luis Potosí, Soochow University. Now, he is an assistant professor in Computer Science at University of Information Science and Technology "St. Paul the Apostle". He is (co)author of more than 120 publications including four books.