American Journal of Applied Sciences

Original Research Paper

Pedestrian Detection in RGB-D Data Using Deep Autoencoders

Pavel Aleksandrovich Kazantsev and Pavel Vyacheslavovich Skribtsov

PAWLIN Technologies Ltd, Dubna, Russia

Article history
Received: 23-09-2015
Revised: 08-10-2015
Accepted: 10-10-2015

Corresponding Author:
Pavel Vyacheslavovich
Skribtsov
PAWLIN Technologies Ltd,
Dubna, Russia
Email: info@pawlin.ru

Abstract: Recent popularity of RGB-D sensors mostly comes from the fact that RGB-images and depth maps supplement each other in machine vision tasks, such as object detection and recognition. This article addresses a problem of RGB and depth data fusion for pedestrian detection. We propose pedestrian detection algorithm that involves fusion of outputs of 2D- and 3D-detectors based on deep autoencoders. Outputs are fused with neural network classifier trained using a dataset which entries are represented by pairs of reconstruction errors of 2D- and 3D-autoencoders. Experimental results show that fusing outputs almost totally eliminate false accepts (precision is 99.8%) and brings recall to 93.2% when tested on the combined dataset that includes a lot of samples with significantly distorted human silhouette. Though we use walking pedestrians as objects of interest, there are few pedestrian-specific processing blocks in this algorithm, so, in general, it can be applied to any type of objects.

Keywords: Pedestrian Detection, Deep Autoencoders, Data Fusion, RGB-D, Image Processing

Introduction

Modern RGB-D sensors typically consisting of optical camera and structured-light depth sensors provide high-quality synchronized videos both in terms of color and depth. Their attractiveness in object detection and recognition tasks comes from the fact that these sensors capture two types of signal that supplement each other in several ways (Cadena and Kosecka, 2013; Collet Romea *et al*., 2011; Lai *et al*., 2011). Indeed, the use of depth maps allows overcoming major difficulties such as variations of texture, illumination, image blur and digital noise. On the other hand, depth maps lack color information and suffer from missing and distorted values caused by infrared-specific noises (Shen and Cheung, 2013). Pedestrian detection falls into a category of object detection algorithms and has been an area of extensive research for more than a decade already (Benenson *et al*., 2015), if we would count from the milestone works by Viola *et al*. (2003) and first attempts to apply histogram of oriented gradients (Lin and Davis, 2008; Felzenszwalb *et al*., 2008) that subsequently became a part of many state-of-art algorithms till today (Park *et al*., 2013; Zhang *et al*., 2014; Benenson *et al*., 2013).

Most state-of-art object detection and recognition systems are still based on handcrafted features, such as SIFT (Lowe, 2004), Spin Images (Johnson and Hebert, 1999), SURF (Bay *et al*., 2008), Fast Point Feature Histogram (Morisset *et al*., 2009), LINE-MOD (Hinterstoisser *et al*., 2011), or feature combinations (Bo *et al*., 2011; Lai *et al*., 2011). Recently introduced approaches to object detection and classification in RGB-D data make use of unsupervised feature learning methods (Blum *et al*., 2012; Bo *et al*., 2012), including deep learning (Socher *et al*., 2012; Lee *et al*., 2015). The latter has become quite popular in recent years, because of outstanding classification results on complicated object datasets represented by RGB images (Krizhevsky *et al*., 2012; Ciresan *et al*., 2012). The effectiveness of deep architectures is commonly accounted for their ability to extract informative feature sets from uncategorized data (Zeiler and Fergus, 2014) as opposed to handcrafted descriptors that need to be redesigned depending on a task and which in formativeness often relies on the developer's expertise. The same reasoning stands behind utilization of deep learning in pedestrian detection on RGB-images (Ouyang and Wang, 2013; Norouzi *et al*., 2009; Sermanet *et al*., 2013), plus contextual information learning (Zeng *et al*., 2013) and occlusion handling (Ouyang and Wang, 2012). Recent methods for pedestrian detection in RGB-D data include multiple-view detector fusion using Markov chains (Choi *et al*.,

2011) and Histogram of Oriented Depths (HOD) inspired by Histogram of Oriented Gradients (HOG) commonly used for pedestrian detection in images (Spinello and Arras, 2011).

Through our extensive research of literature dedicated to pedestrian detection in RGB-D data we could not find any works that utilize deep learning methods. Speculating on this matter one could have come to the assumption that this may be caused by a mixture of the following reasons. First reason is a lack of vast RGB-D datasets of pedestrians. In comparison to a number of world-wide recognized benchmark datasets in the area of pedestrian detection in RGB-images (Benenson *et al*., 2015), a few representatives of RGB-D datasets, like the one presented in (Borràs *et al*., 2012), look like a drop in the sea. Moreover, these RGB-D datasets do not bear benchmark status that feels quite discouraging and impose extra difficulties on a research process, like filming datasets with complex additional samples (occlusions, outdoor scenes, distortions of human silhouette, etc.) (Spinello and Arras, 2011), which usually are present in state-of-art RGB benchmarks. Also, the later actually became widely possible only with recent introduction of cheap and easy-to-use RBG-D sensors, like Microsoft Kinect for Windows. The second reason is that modern infrared 3D-sensors have a range of few meters-in database presented in (Borràs *et al*., 2012), depth values of pedestrians start to “fade” after 4 meters mark. Currently, this fact limits application of pedestrian detectors in RGB-D data to indoor use. The third reason could be a novelty of deep learning methods, so not every research team has added them to their developer toolset yet. We dare to claim that this work is a first published attempt to apply deep learning method to pedestrian detection in RGB-D data.

We make few assumptions that are not explicitly made in other works dedicated to pedestrian detection and, in general, object detection in RGB-D data. First, we assume that one type of the signal may not be available at any given time. It is indeed possible in real-world applications where RGB-image can be void due to lack of illumination, or where depth values of pedestrian can be severely distorted, because he or she wears light-absorbing clothes-objects with darker colors, specular surfaces, or fine-grained surfaces like human hair are prime candidates for poor depth measurements (Cho *et al*., 2008). Most works dedicated to object detection and classification in RBG-D data, like (Blum *et al*., 2012; Bo *et al*., 2012; Socher *et al*., 2012; Lee *et al*., 2015), rely on both RBG-image and depth map by fusing data modes on feature level, i.e., combining feature vectors from features extracted from both RGB and depth channels. This approach will lead to malfunction of the system based on it, if one of the signals is missing. Second, we assume that pedestrians can carry backpacks, cases, wheel-bags and outer clothing (including fur-coats, hats, hoods, etc.) and other accessories that severely distort human silhouette. Last assumption simplifying our research states that we do not consider occlusions of upper body.

In summary, the contributions of this paper are:

- We propose deep learning algorithm based on autoencoders for pedestrian detection in RGB-D data. To the best of our knowledge this is a first work to do so
- In order to handle shifts of human silhouettes inside regions of interest, preemptively extracted via segmentation, we insert a unique regularization terms that allow to explicitly separate components of encoding vector into two categories: Class attributes representation (the p-parameters) and transformation attributes (q-parameters). This separation allows using only class attributes representation for reconstruction

Methodology

In this article we propose pedestrian detection algorithm that involves fusion of outputs (reconstruction errors) of 2D- and 3D-detectors that are based on deep autoencoders. Fusion is done by finding a precise correspondence between RGB-image and depth map and subsequent neural network classification of two-component vector that consists from reconstruction errors of 2D- and 3D-detectors. Outputs are fused with neural network classifier trained using a training set which entries are represented by pairs of the reconstruction errors of 2D- and 3D-detectors. An optional tracking can be used to assign labels to objects in video-sequence. Forward pass scheme of our algorithm is shown in Fig. 1.

Preprocessing and Segmentation

Preprocessing part is rather simple and includes depth map and image downscaling followed by conversion to grayscale. Segmentation part is used to significantly reduce the number of candidate windows (in this study we will refer them as *regions of interest*) in initial image or depth map. For this purpose we use MSER algorithm (maximally stable extremal regions) (Matas *et al*., 2002). In order to avoid potential loss of pedestrians in the depth maps and images we tune down threshold so that we increase a number of extremal regions, before they get merged into maximally stable extremal regions via component tree algorithm. Also, we apply MSER algorithm to both intensity and inverse image/depth map. MSERs then get approximated by ellipses. Some of the ellipses are filtered out by angle of deviation from vertical axis, since pedestrians are vertical objects by nature.

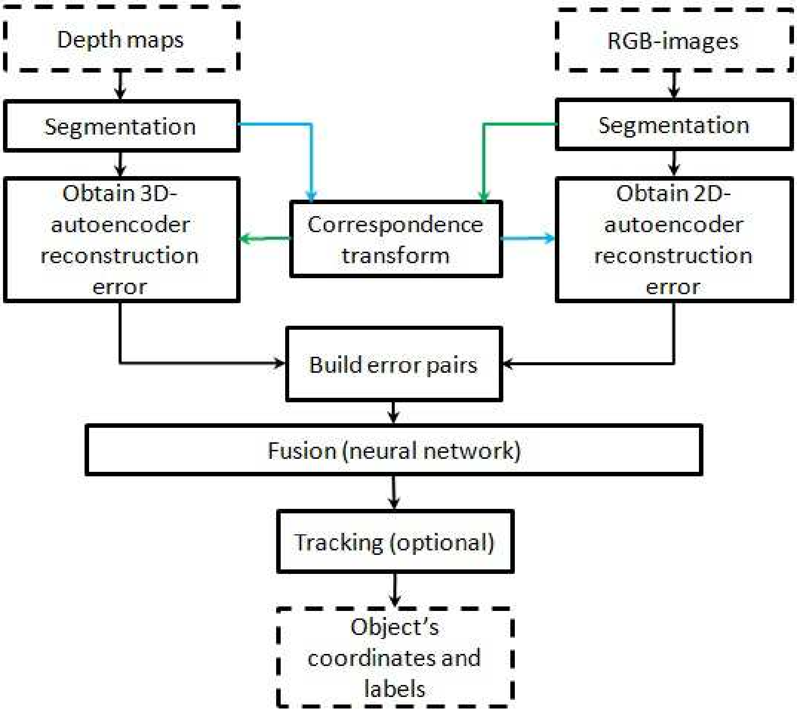


Fig. 1. Forward pass scheme of our algorithm

Angle threshold is set to 20° to account for inaccuracies of MSERs detection. Remaining ellipses are then approximated by bounding vertical rectangles which get their bottom cut off. This is done in order to remove lower body from equation, since this part of pedestrian silhouette is quite variable. Cut-off parameter is experimentally defined as 0.65. Segmentation procedure is illustrated in Fig. 2.

Inaccuracies of MSER segmentation lead to shifts of upper body inside of the remaining bounding rectangles in all directions. This issue is addressed by including shifted samples into the training set and introducing special regularization procedure, described in subsection 2.2. Rectangles found in RGB image are mapped to depth map and vice versa, using direct and inverse transforms. All rectangles are rescaled to a fixed size and their contents are fed to autoencoders. Transform parameters are calculated beforehand using our algorithm which would take another article to cover it. Some hints can be given here, though. Our calibration algorithm is motivated by (Han and Bhanu, 2007), where we got our transform model, but our model fitting algorithm is different from (Han and Bhanu, 2007) and uses reconstruction errors of 2D- and 3D-detectors to build pairs of hypothesis that are subsequently fed to RANSAC algorithm.

Deep Autoencoders Training

In order to classify rectangle segments we use deep autoencoders. Deep autoencoders are multilayer auto-associative neural networks trained by iterative procedure that is commonly referred to as “Deep Learning” (Krizhevsky and Hinton, 2011; Baldi, 2012) that can effectively reduce the dimensions of the original signal and generate non-local higher-level attributes of objects. The basics of autoencoders training for object detection are described very well in literature (Szegedy *et al*., 2013), so here we focus on two important parts of algorithm which we contributed to-regularization and reconstruction error calculation.

Regularization

There are three most commonly used regularization procedures: Weight decay (Bishop, 2007), penalization of output sensitivity to input, measured as Frobenius norm of the Jacobian (Rifai *et al*., 2011), sparsity regularization using Kullback–Leibler divergence (Yu *et al*., 2013). Here we present new regularization procedure that make it possible to separate encoding vector (representation) into two components in an explicit form, that is, to the class attributes representation (the p-parameters) and transformation attributes (q-parameters). In this article this regularization procedure will be referred to as *pq-regularization*. Figures 3 and 4 illustrate this approach. Knowing which neurons represent class features we can discard q-part that encodes transformation and use only p-part that encodes object.

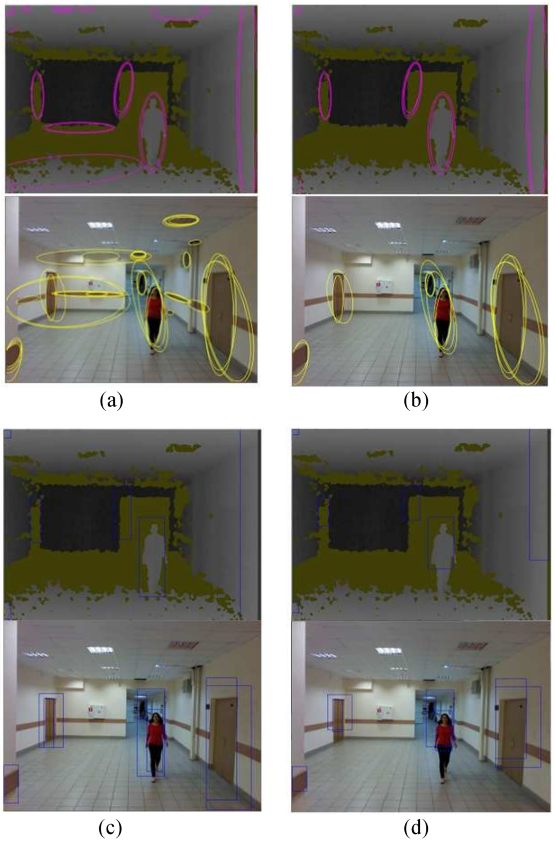


Fig. 2. Segmentation process (upper row-depth map, lower row-images): (a) MSER detection and approximation by ellipses (b) filtering ellipses by angle of deviation from vertical axis (c) approximating remaining ellipses by bounding rectangles (d) removing

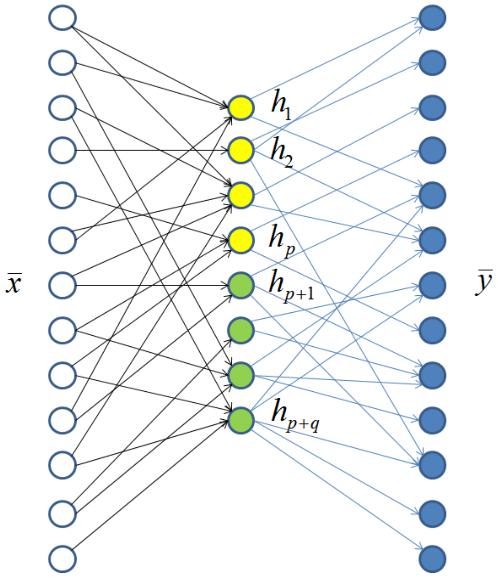


Fig. 3. Autoencoder with the encoding layer *h*, the first *p*-component of which is responsible for object's class features and the following *q*-component-for object's transformation

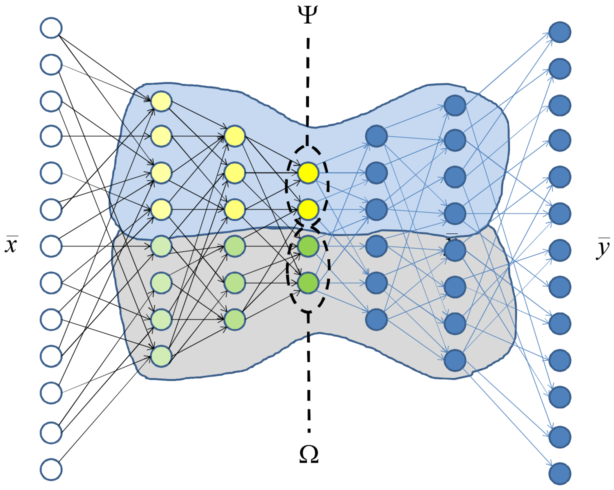


Fig. 4. Multi-layer autoencoder, in which the input representation separation from a layer to a layer is enhanced, ideally, when reaching the complete separation of information about the class and transformation of the objects at the last level of the encoder

Autoencoder with one hidden layer may be described by the expressions:

$$\bar{y}(\bar{x},\bar{w}) = \phi\left(D\bar{h}\right)\ \bar{h} = \phi\left(H\bar{x}\right) \qquad (1)$$

where, $\bar{x} \in R^N$ is an input signal; $\bar{y} \in R^N$ is an output signal of the network with the same dimension as the input signal; $\bar{w}$ are weight coefficients of the neural network comprising of encoding layer matrix coefficients; $H \in R^{(p+q)\times N}$ and decoding layer matrix $D \in R^{(p+q)\times N}$, as well as the neuron shifts weights, φ is the element-wise nonlinear vector activation function (offsets are omitted for the purposes of shortening). In this study we use symmetric sigmoid activation function $\phi(x) = \frac{1}{1+\exp(-x)} - \frac{1}{2}$.

The intermediate output of the network encoding (encoder) can be described by the vector $\bar{h}(\bar{x},H)$. In this case, we assume that the first p-components of this vector shall be responsible for the object class information and the following q-components for information about the transformations (shifts). Autoencoder training is done by solving the optimization problem with additional regularizing components:

$$\bar{w}^* = \arg\min_{\bar{w}} \left\{ \sum_i E_i + T_1(\lambda,\bar{w}) + T_2(\alpha,\bar{w}) + T_3(\beta,\bar{w}) \right\} \qquad (2)$$

where, w^* are the required weight coefficients of the neural network; $\{\bar{x}_i, c_i\}$-training examples ($\bar{x}_i \in R^N$ is an attribute vector, c_i is class number) to the task of recognizing objects in the image; θ-positive constant, λ,α,β ∈ R are regularization coefficients. $\bar{h}^p = \left(h_1 \quad h_2 \quad \ldots \quad h_p\right)$ and $\bar{h}^q = \left(h_{p+1} \quad h_{p+2} \quad \ldots \quad h_{p+q}\right)$ -p-, q-components of the encoding vector.

$T_1(\lambda,\bar{w}) = \lambda\|\bar{w}\|^2$ -regularization of the representation 'simplicity' (in this case using weight decay (Bishop, 2007)):

$$T_2(\alpha,\bar{w}) = \alpha \sum_{\forall i,j:c_i=c_j} \left\|\bar{h}^p(\bar{x}_i,\bar{w}) - \bar{h}^p(\bar{x}_j,\bar{w})\right\|^2 \quad (3)$$

Consistency regularization of the p-components of representation in the hidden layer. This term penalizes a current state of the autoencoder if p-components yield different outputs for a pair of same-class samples, even if they have different transformations:

$$T_3(\beta,\bar{w}) = \beta \sum_{\forall i,j:\omega(\bar{x}_i,\bar{x}_j)>\theta} \omega(\bar{x}_i,\bar{x}_j)\left\|\bar{h}^q(\bar{x}_i,\bar{w}) - \bar{h}^q(\bar{x}_j,\bar{w})\right\|^2 \quad (4)$$

Consistency regularization of the q-representation. This term penalizes a current state of the autoencoder if q-components yield different outputs for a pair of same-transformation samples, even if they have different classes. $\omega(\bar{x}_i,\bar{x}_j)$ is object transformation similarity measure evaluation function. Small values of the function mean the maximum difference of transformation, large values mean transformations matches.

There are total of six possible transitions considered: Shift left, shift right, shift up, shift down, compression along the horizontal axis, compression along the vertical axis. That is why in this particular task a number of q-parameters is limited to six.

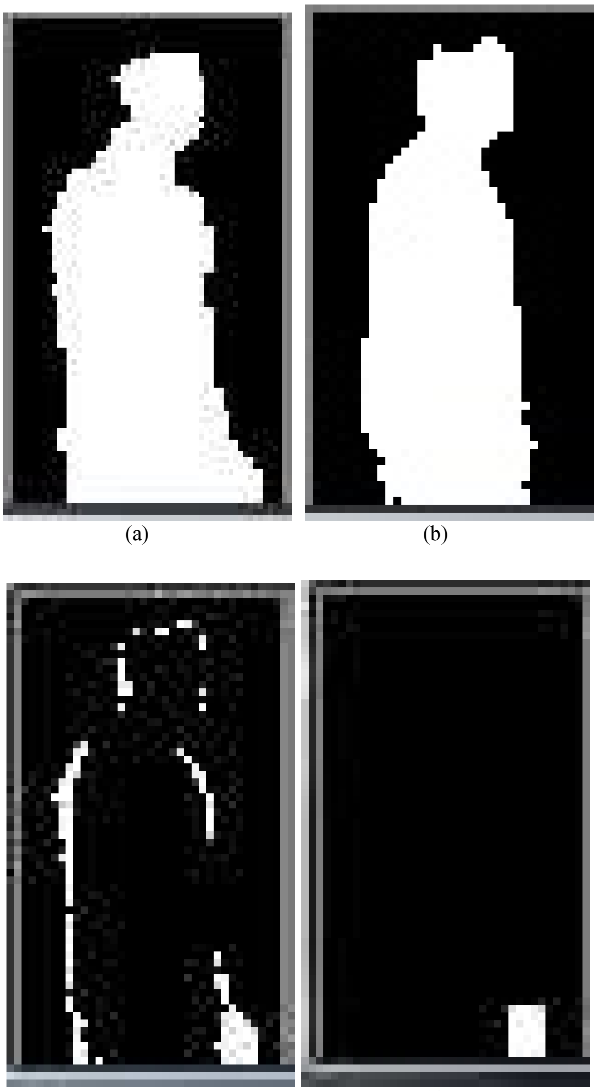
There are total of eight classes of objects considered. These classes correspond to movement directions of pedestrians, i.e.,: Left to right, right to left, to the camera, out off camera, diagonal left to right to the camera, diagonal left to right out off camera, diagonal right to left to the camera, diagonal right to left out off camera. Train pairs for evaluation of T_2 and T_3 terms were generated with exact information about classes and transformations generated automatically from original non-transformed samples taken from the training set.

E_i is reconstruction error. Usually it is calculated as a distance between input and output using L_1 or L_2 metrics. We use different approach to its calculation described below.

Reconstruction Error

Calculation of reconstruction errors as distances between input and output using L_1 or L_2 metrics has a significant drawback. This drawback comes from the fact that even small discrepancies between input and reconstructed patterns (especially at the object edges) yield significant error values. In case of L_1 metric this error grows linearly, in case of L_2 metric-quadratically. It is a common case when an object was reconstructed well, but even a thin line of discrepancy along the reconstructed object's border yields inequitably high, as illustrated in Fig. 5.

In order to remove that kind of discrepancy we, first, binarize input and output, take its L_1-difference and then apply morphological opening operation with a rectangle kernel. This operation is proven to be effective for removal of thin lines and small outliers. An example of its application is shown in Fig. 5d. Final reconstruction error is obtained by calculating L_2 norm of morphologically opened difference between binarized input and output.



(a) (b)

(c) (d)

Fig. 5. Discrepancies at reconstructed object's boundaries (a) binarized inpu; (b) binarized output; (c) binarized difference between input and output; (d) binarized difference between input and output after morphological opening

2D- and 3D-Detectors Outputs Fusion

Pairs of reconstruction errors are fed to another neural network as inputs. This neural network fuses a pair of reconstruction errors into one single output that assumes floating point values in a range of [0.1] with 0 corresponding to "non-object" event and 1 - to "object" event. Fusion neural network (FNN for further use in this article) has only one hidden layer and is trained using a training set that consists of reconstruction error pairs obtained by 2D- and 3D-detectors during their forward-pass on a set of 456 image-depth map pairs. 1088 training vectors were cropped from these 456 pairs. Training vectors were formed in 8 categories. Each category corresponds to a pair of events. These events are listed in Table 1.

Table 2 lists categories of training samples, a number of training samples in each category and desired output for each category.

Datasets

Our training and verification sets were combined of various datasets available in the Internet (like DGait dataset (Borràs *et al*., 2012) for pedestrian samples and Berkeley 3-D Object Dataset (Janoch *et al*., 2011) for non-pedestrian samples) and a dataset filmed by ourselves. All datasets used were captured with Microsoft Kinect for Windows. The dataset filmed by ourselves includes in-door and out-door records with pedestrians walking in different directions. In our dataset we focus on pedestrians that carry backpacks, cases, wheel-bags and heavy outer clothing (including fur-coats, hats, hoods, etc.) and other accessories that severely distort human silhouette. Pedestrians were marked-up manually in both video and depth records, thus providing us with exact coordinates of positive samples. Combined database included 3,546 unique pedestrian samples more or less evenly distributed among the categories of pedestrian movement directions. However, the initial amount of positive samples was multiplied by shifts and compression along the vertical and horizontal axis. Shift and compression values were experimentally defined by observing inaccuracies of MSER segmentation. Total of 10 variations were applied to each initial sample providing us with 35,460 of positive samples overall. A subset of negative samples was obtained by running our segmentation algorithm first on all databases in our disposal (including databases that do not contain pedestrians at all) and then excluding MSER-detected pedestrians. The total of negative samples collected this way is slightly over 10,000. Training and verification datasets do not overlap.

Table 1. Notations of events used to form categories of training vectors for FNN

Event	Notation
3D true accept	3DTA
3D false accept	3DFA
3D false reject	3DFR
3D true reject	3DTR
2D true accept	2DTA
2D false accept	2DFA
2D false reject	2DFR
2D true reject	2DTR

Table 2. Categories and number of training samples for FNN

Category (pair of events)	Number of samples	Desired output
3DTA-2DTA	115	1
3DTA-2DFR	287	1
3DFA-2DTR	74	0
3DFA-2DFA	32	0
3DFR-2DTA	98	1
3DFR-2DFR	89	1
3DTR-2DFA	342	0
3DTR-2DTR	51	0

Results

In order to obtain optimal parameters for autoencoders structure and regularization coefficients λ,α,β (see expressions (2)-(4)) we ran a set of experiments in accordance with Table 3 and assuming following limitations:

- A number of neurons in a next encoding layer must be less than in the previous one
- Six neurons in each encoding layer are reserved for transformation representation, so effectively each layer must have no less than 10 neurons in total
- The sum of regularization coefficients should not exceed 1

Table 4 shows optimal parameters for 2D- and 3D-autoencoders, respectively. Number of inputs for both autoencoders is set to 2,170 and corresponds to 35x62 rectangle fragment, so only hidden layer structure is presented. Optimal set of parameters corresponds to the largest F-measure value.

Table 5 shows accuracy results for 2D- and 3D-autoencoders tested separately.

Table 3. Ranges and steps of variable parameters of autoencoders and fusion neural network

Variable parameter	Ranges of variation	Step
No. of 1st hidden layer neurons	from 500 to 100	5
No. of 2nd hidden layer neurons	from 50 to 30	2
No. of 3rd hidden layer neurons	from 20 to 10	1
λ	from 0.05 to 0.5	0.05
α	from 0.05 to 0.5	0.05
β	from 0.05 to 0.5	0.05
No. of FNN hidden layer neurons	from 15 to 5	1
FNN threshold	from 0.6 to 0.99	0.01

Discussion

Analyzing the results presented in Table 5 we can draw a conclusion that 2D- and 3D-autoencoders mutually supplement each other very well. Figure 6 illustrates some of the detection results. Numerical accuracy results are better than those shown in (Spinello and Arras, 2011)-at recall rate of 90% they have precision of 80%. Observing false detects of our autoencoders on test data we could conclude that in majority of cases, when 2D-autoencoder yields a false accept, 3D-detector will yield the true reject and vice-versa. In other words, 2D- and 3D-detector's false accepts are almost mutually exclusive. Spinello and Arras (2011), however, partial occlusions can be handled (though the number of occluded samples in database is not provided). Our algorithm overall fails to separate overlapping pedestrians, usually detecting them as one body. But we have not counted these cases in accuracy results, since initially we did not intended to handle them. Our algorithm, however, can handle severe distortions of human silhouettes (outer-clothing, bags, back-packs, etc.) and performs outdoors just as fine as indoors. Also, (Spinello and Arras, 2011) is quite computationally heavy and able to run in real time only using GPU-acceleration (30 fps), as the authors state. Our algorithm can run on 5 fps without any GPU-acceleration. But since autoencoder is a neural network essentially, we will be able to speed it up on GPU due to its inherent massive parallelism.

Table 4. Set of optimal parameters and results obtained with optimal set of parameters

Parameters	Results
2D-auto encoder	46×24
3D-auto encoder	112×36×12
λ (2D/3D)	0.15/0.15
α (2D/3D)	0.35/0.40
β (2D/3D)	0.50/0.45
No. of FNN hidden layer neurons	8
FNN threshold	0.96
Precision	99.8%
Recall	93.2%
F-measure	96.48%

Table 5. Accuracy results for 2D- and 3D-autoencoders tested separately

	Precision (%)	Recall (%)
2D	90.6	81.7
3D	96.8	85.2
2D+3D	99.8	93.2

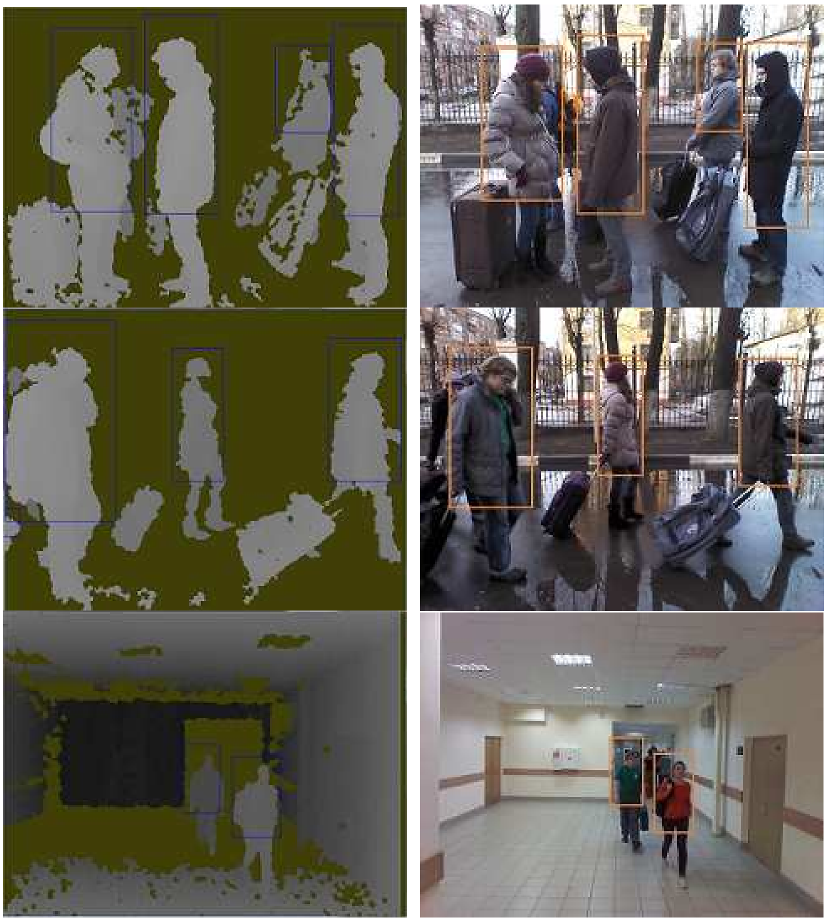


Fig. 6. Proposed algorithm works well in outdoor conditions (outer clothing, depth sensor noises, non-uniform background)

Conclusion

In this study we have introduced a new approach to pedestrian detection in RGB-data based on deep autoencoders. 2D- and 3D-autoencoders yield average results when used separately, but fusing their reconstruction errors using another neural network yields outstanding results (93.2% recall at 99.8% precision). A combined detector is able to detect pedestrians that carry backpacks, cases, wheel-bags and outer clothing (including fur-coats, hats, hoods, etc.) and other accessories that severely distort human silhouette. MSER-based segmentation algorithm roughly extracts regions of interest that contain pedestrians, but with significant shifts and compressions along one of the axis. These transformations are dealt with by novel regularization procedure during autoencoder training process. The algorithm runs at *5 fps* without any hardware acceleration.

Effective shift and compression handling using novel deep autoencoder regularization procedure allows to roughly segment RGB-D image and analyze much less regions than it is required by HOG-based algorithms. Reconstruction errors of 2D- and 3D-autoencoders seem to be very informative as inputs for fusion neural network, that almost totally eliminates false detects obtained by 2D- and 3D-detectors used separately.

Authors would dare to assume, that this article may show a possible strategy for pedestrian RGB-D detection algorithms development that enables to part with computationally expensive HOG-based methods and still keep accuracy numbers high.

Acknowledgement

The authors thank Ministry of Education and Science of the Russian Federation for supporing this research.

Funding Information

The research was financed by the Ministry of Education and Science of the Russian Federation under the Grant Contract dated 24 November, 2014, (grant number 14.579.21.0070, contract unique identifier RFMEFI57914X0070), the grant was assigned for the research into “2D- and 3D-data fusion and processing software for object detection and recognition”.

Author’s Contributions

Authors contributed to this research as follows:

Pavel Aleksandrovich Kazantsev: Designed research plan and algortihm flow diagram, developed detection and fusion algorithms, carried out validation and tests.

Pavel Vyacheslavovich Skribtsov: Developed regularization procedure and autoencoder’s training alogirthm, organized collection of training and validation datasets, dicussed the results.

Ethics

The authors have no conflicts of interest in the development and publication of current research.

References

Janoch, A., S. Karayev, Y. Jia, J.T. Barron and M. Fritz *et al*., 2011. B3DO: Berkeley 3-D object dataset.

Baldi, P., 2012. Autoencoders, unsupervised learning and deep architectures. Proceedings of the JMLR: Workshop on Unsupervised and Transfer Learning, (UTL’ 12) pp: 37-50.

Bay, H., A. Ess, T. Tuytelaars and L. Van Gool, 2008. Speeded-Up Robust Features (SURF). Comput. Vision Image Understand., 110: 346-359. DOI: 10.1016/j.cviu.2007.09.014

Benenson, R., M. Omran, J. Hosang and B. Schiele, 2015. Ten Years of Pedestrian Detection, what have we Learned? Computer Vision-ECCV 2014 Workshops, Agapito, L., M.M. Bronstein and C. Rother (Eds.), Springer, ISBN-10: 978-3-319-16180-8, pp: 613-627.

Benenson, R., M. Mathias, T. Tuytelaars and L. Van Gool, 2013. Seeking the strongest rigid detector. Proceedings of the Conference on Computer Vision and Pattern Recognition, Jun. 23-28, IEEE Xplore Press, Portland, OR, pp: 3666-3673. DOI: 10.1109/CVPR.2013.470

Bishop, C.M., 2007. Pattern Recognition and Machine Learning. 1st Edn., Springer, New York, ISBN-10: 0387310738, pp: 738.

Blum, M., J.T. Springenberg, J. Wifing and M. Riedmiller, 2012. A learned feature descriptor for object recognition in RGB-D data. Proceedings of the International Conference on Robotics and Automation, May 14-18, IEEE Xplore Press, Saint Paul, MN, pp: 1298-1303. DOI: 10.1109/ICRA.2012.6225188

Bo, L., X. Ren and D. Fox, 2011. Depth kernel descriptors for object recognition. Proceedings of the International Conference on Intelligent Robots and Systems, Sept. 25-30, IEEE Xplore Press, San Francisco, CA, pp: 821-826. DOI: 10.1109/IROS.2011.6095119

Bo, L., X. Ren and D. Fox, 2012. Unsupervised feature learning for RGB-D based object recognition. Proceedings of the 13th International Symposium on Experimental Robotics, (SER’ 12), Springer, pp: 387-402. DOI: 10.1007/978-3-319-00065-7_27

Borràs, R., A. Lapedriza and L. Igual, 2012. Depth information in human gait analysis: An experimental study on gender recognition. Proceedings of the 9th International Conference on Image Analysis and Recognition, (LAR’ 12), Springer-Verlag Berlin, pp: 98-105. DOI: 10.1007/978-3-642-31298-4_12

Cadena, C. and J. Kosecka, 2013. Semantic parsing for priming object detection in RGB-D scenes. Proceedings of the 3rd Workshop on Semantic Perception, Mapping and Exploration, (PME' 13), SPME, pp: 1-6. DOI: 10.1177/0278364914549488

Cho, J., S. Kim, Y. Ho and K. Lee, 2008. Dynamic 3D human actor generation method using a time-of-flight depth camera. IEEE Trans. Consumer Electron., 54: 1514-1521. DOI: 10.1109/TCE.2008.4711195

Choi, W., C. Pantofaru and S. Savarese, 2011. Detecting and tracking people using an rgb-d camera via multiple detector fusion. Proceedings of the International Conference on Computer Vision Workshops, Nov. 6-13, IEEE Xplore Press, Barcelona, pp: 1076-1083. DOI: 10.1109/ICCVW.2011.6130370

Ciresan, D., U. Meier and J. Schmidhuber, 2012. Multi-column deep neural networks for image classification. Proceedings of the Conference on Computer Vision and Pattern Recognition, Jun. 16-21, IEEE Xplore Press, Providence, RI, pp: 3642-3649. DOI: 10.1109/CVPR.2012.6248110

Collet Romea, A., M. Martinez Torres and S. Srinivasa, 2011. The moped framework: Object recognition and pose estimation for manipulation. Int. J. Robot. Res., 30: 1284-1306. DOI: 10.1177/0278364911401765

Felzenszwalb, P., D. McAllester and D. Ramanan, 2008. A discriminatively trained, multiscale, deformable part model. Proceedings of the Conference on Computer Vision and Pattern Recognition, Jun. 23-28, IEEE Xplore Press, Anchorage, AK, pp: 1-8. DOI: 10.1109/CVPR.2008.4587597

Han, J. and B. Bhanu, 2007. Fusion of color and infrared video for moving human detection. Patt. Recognit., 40: 1771-1784. DOI: 10.1016/j.patcog.2006.11.010

Hinterstoisser, S., S. Holzer, C. Cagniart, S. Ilic and K. Konolige *et al*., 2011. Multimodal templates for real-time detection of texture-less objects in heavily cluttered scenes. Proceedings of the International Conference on Computer Vision, Nov. 6-13, IEEE Xplore Press, Barcelona, pp: 858-865. DOI: 10.1109/ICCV.2011.6126326

Johnson, A. and M. Hebert, 1999. Using spin images for efficient object recognition in cluttered 3D scenes. Trans. Patt. Analysis Machine Intellig., 21: 433-449. DOI: 10.1109/34.765655

Shen, J. and S.C.S. Cheung, 2013. Layer depth denoising and completion for structured-light rgb-d cameras. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Jun. 23-28, IEEE Xplore Press, Portland, OR, pp: 1187-1194. DOI: 10.1109/CVPR.2013.157

Krizhevsky, A. and G.E. Hinton, 2011. Using very deep autoencoders for content-based image retrieval. BibSonomy.

Krizhevsky, A., I. Sutskever and G.E. Hinton, 2012. Imagenet Classification with Deep Convolutional Neural Networks. In: Advances in Neural Information Processing Systems 25: 26th Annual Conference on Neural Information Processing Systems, Pereira, F., C.J.C. Burges, L. Bottou and K.Q. Weinberger (Eds.), Red Hook, NY, ISBN-10: 162748003X, pp: 1097-1105.

Lai, K., L. Bo, X. Ren and D. Fox, 2011. A large-scale hierarchical multi-view RGB-D object dataset. Proceedings of the International Conference on Robotics and Automation, May 9-13, IEEE Xplore Press, Shanghai, pp: 1817-1824. DOI: 10.1109/ICRA.2011.5980382

Lee, H., I. Lenz and A. Saxena, 2015. Deep learning for detecting robotic grasps. Int. J. Robot. Res., 34: 705-724. DOI: 10.1177/0278364914549607

Lin, Z.L. and L.S. Davis, 2008. A pose-invariant descriptor for human detection and segmentation. In: Computer Vision-ECCV, Forsyth, D., P. Torr and A. Zisserman (Eds.), Springer, ISBN-10: 978-3-540-88692-1, pp: 423-436.

Lowe, D., 2004. Distinctive image features from scale-invariant keypoints. Int. J. Comput. Vision, 60: 91-110. DOI: 10.1023/B:VISI.0000029664.99615.94

Matas, J., O. Chum, M. Urban and T. Pajdla, 2002. Robust wide-baseline stereo from maximally stable extremal regions. Image Vision Comput., 22: 384-393. DOI: 10.1016/j.imavis.2004.02.006

Morisset, B., R. Bogdan Rusu, A. Sundaresan and K. Hauser *et al*., 2009. Leaving flatland: Toward real-time 3D navigation. Proceedings of the International Conference on Robotics and Automation, May 12-17, IEEE Xplore Press, Kobe, pp: 3786-3793. DOI: 10.1109/ROBOT.2009.5152715

Norouzi, M., M. Ranjbar and G. Mori, 2009. Stacks of convolutional restricted Boltzmann machines for shift-invariant feature learning. IEEE Conference on Computer Vision and Pattern Recognition, Jun. 20-25, IEEE Xplore Press, Miami, FL, pp: 2735-2742. DOI: 10.1109/CVPR.2009.5206577

Ouyang, W. and X. Wang, 2012. A discriminative deep model for pedestrian detection with occlusion handling. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Jun. 16-21, IEEE Xplore Press, Providence, RI, pp: 3258-3265. DOI: 10.1109/CVPR.2012.6248062

Ouyang, W. and X. Wang, 2013. Joint deep learning for pedestrian detection. Proceedings of the IEEE International Conference on Computer Vision, Dec. 1-8, IEEE Xplore Press, Sydney, VIC, pp: 2056-2063. DOI: 10.1109/ICCV.2013.257

Park, D., C.L. Zitnick, D. Ramanan and P. Dollár, 2013. Exploring weak stabilization for motion feature extraction. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Jun. 23-28, IEEE Xplore Press, Portland, OR, pp: 2882-2889. DOI: 10.1109/CVPR.2013.371

Rifai, S., P. Vincent, X. Muller, X. Glorot and Y. Bengio, 2011. Contracting auto-encoders: Explicit invariance during feature extraction. Proceedings of the 28th International Conference on Machine Learning, (CML' 11), pp: 833-840.

Sermanet, P., K. Kavukcuoglu, S. Chintala and Y. Lecun, 2013. Pedestrian detection with unsupervised multi-stage feature learning. Proceedings of the 2013 IEEE Conference on Computer Vision and Pattern Recognition, (VPR' 13), IEEE Computer Society Washington, DC, USA, pp: 3626-3633. DOI: 10.1109/CVPR.2013.465

Socher, R., B. Huval, B.P. Bath, C.D. Manning and A.Y. Ng, 2012. Convolutional-Recursive Deep Learning for 3D Object Classification. In: Advances in Neural Information Processing Systems 25: 26th Annual Conference on Neural Information Processing Systems, Pereira, F., C.J.C. Burges, L. Bottou and K.Q. Weinberger (Eds.), Red Hook, NY, ISBN-10: 162748003X, pp: 665-673.

Spinello, L. and K. Arras, 2011. People detection in RGB-D data. Proceedings of the International Conference on Intelligent Robots and Systems, Sept. 25-30, IEEE Xplore Press, San Francisco, CA, pp: 3838-3843. DOI: 10.1109/IROS.2011.6095074

Szegedy, C., A. Toshev and D. Erhan, 2013. Deep Neural Networks for Object Detection. Advances in Neural Information Processing Systems 25: 26th Annual Conference on Neural Information Processing Systems, Pereira, F., C.J.C. Burges, L. Bottou and K.Q. Weinberger (Eds.), Red Hook, NY, ISBN-10: 162748003X, pp: 2553-2561.

Viola, P., M.J. Jones and D. Snow. 2003. Detecting pedestrians using patterns of motion and appearance. Proceedings of the 9th IEEE International Conference on Computer Vision, Oct. 13-16, IEEE Xplore Press, Nice, France, pp: 734-741. DOI: 10.1109/ICCV.2003.1238422

Yu, D., K. Yao, H. Su, G. Li and F. Seide, 2013. KL-divergence regularized deep neural network adaptation for improved large vocabulary speech recognition. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, May 26-31, IEEE Xplore Press, Vancouver, BC, pp: 7893-7897. DOI: 10.1109/ICASSP.2013.6639201

Zeiler, M.D. and R. Fergus, 2014. Visualizing and Understanding Convolutional Networks. In: Computer Vision-ECCV, Forsyth, D., P. Torr and A. Zisserman (Eds.), Springer, ISBN-10: 978-3-540-88692-1, pp: 818-833.

Zeng, X., W. Ouyang and X. Wang, 2013. Multi-stage contextual deep learning for pedestrian detection. Proceedings of the IEEE International Conference on Computer Vision, Dec. 1-8, IEEE Xplore Press, Sydney, VIC, pp: 121-128. DOI: 10.1109/ICCV.2013.22

Zhang, S., C. Bauckhage and A.B. Cremers, 2014. Informed haar-like features improve pedestrian detection. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Jun. 23-28, IEEE Xplore Press, Columbus, OH, pp: 947-954. DOI: 10.1109/CVPR.2014.126