Assembly Defect Detection of Atomizers Based on Machine Vision

Jiankun Wang and Hong Hu
Department of Mechatronic Engineering
Harbin Institute of Technology, Shenzhen
Shenzhen, Guangdong Province, China
honghu@hit.edu.cn

Long Chen and Caiying He
Technology Center
Shenzhen Colibri Technologies Co. Ltd
Shenzhen, Guangdong Province, China
longchen@colibri.com.cn

ABSTRACT

Atomizers are assembled in an automated assembly line, which inevitably creates assembly defects. In this paper, we use machine vision technology to detect assembly defects in atomizers. We propose two algorithms: an image processing algorithm, and a deep learning algorithm based on convolutional neural network. For design of the image processing algorithm, we set the region of interest for detection according to the position of different assembly defects. For the deep learning algorithm, we adopt the MobileNet model and propose a new training program to improve detection accuracy. The paper also includes an evaluation of the performance of the two algorithms and analyzes their advantages and disadvantages.

KEYWORDS

Atomizer, Assembly defect detection, Convolutional neural network, Machine vision

1 Introduction

In industrial automated production, in order to ensure the quality of the product it is necessary to perform defect detection in order to eliminate defective products. Machine vision is often used for defect detection, which increases the automation of production and reduces labor costs. At present, there are two types of algorithms for machine vision, one is a digital image processing algorithm, and the other is a deep learning algorithm based on convolutional neural network (CNN).

In the field of industrial inspection, image processing algorithms are mainly used because the theory of image processing algorithms is more mature. Scholars have designed corresponding image processing detection algorithms for assembly defects of different products.

Jiancheng [1] used the distance measurement method to measure the position of a syringe part in order to detect whether the syringe had been assembled correctly. Jing et al. [2] used the modified

Hausdorff distance matching algorithm to detect the position of a syringe part. Ardhy et al. [3] pre-processed an image using an adaptive Gaussian threshold method, and then performed a differential operation on the standard image and the image to be detected, to detect whether a printed circuit board was defective.

In recent years, deep learning technology has made remarkable achievements in the field of image recognition. Image recognition algorithms based on CNN have been successfully applied in many fields. In the field of industrial defect detection, some scholars have also begun to use CNN for defect detection and classification.

Je-Kang Park et al. [4] designed a simple CNN network structure to detect surface defects of different items. Wu Tong [5] used the X-ray imaging system to collect images of the products and label them, extract the feature of the parts using a CNN, and then train the deep learning model. The model was used to categorize the internal parts of the assembly to detect missing parts.
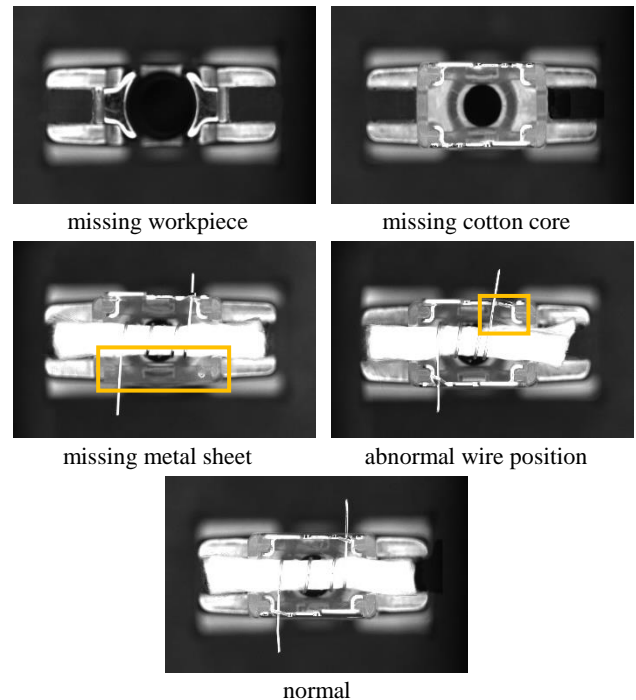

missing workpiece   missing cotton core

missing metal sheet   abnormal wire position

normal

Figure 1: Images of atomizer assembly defects samples

The research object of this paper is an atomizer. The assembly process of an atomizer requires several processes. Failure of the gripping of components can result in missing parts. Machine

vibration and assembly accuracy can make the assembly position of the parts inaccurate. Combined with the actual situation of production, the assembly defects of an atomizer can be divided into four types: missing workpiece, missing cotton core, missing metal sheet, and abnormal wire position. Defect samples are shown in Figure 1.

Since there has been no research on the detection of atomizer assembly defects, this paper proposes two algorithms for detecting atomizer assembly defects. One is the image processing detection algorithm, and the other is the deep learning detection algorithm based on CNN. Our goal is to verify the feasibility of using deep learning algorithms for industrial inspection and to analyze the characteristics of two different algorithms.

2 Image Processing Algorithm

2.1 Detection Target Location and ROI Setting

The first step of detection is to locate the target. Since the relative position of the assembly and fixture is known, we can locate the fixture position first, and then set the corresponding ROI (region of interest) according to the position of the defect in the assembly. The specific steps are as follows.

Step-1: Separate the foreground and background. The image has distinct foreground and background, and can be segmented by image binarization to get the fixture and assembly area. Because the background of the image is black, we use fixed threshold binary segmentation. The formula is as follows:

$$g(x, y) = \begin{cases} 255 & f(x, y) > 60 \\ 0 & f(x, y) \leq 60 \end{cases}$$

Where $g(x, y)$ represents the original pixel value at (x, y), $f(x, y)$ represents the pixel value at (x, y) after binarization.

original image   binarization
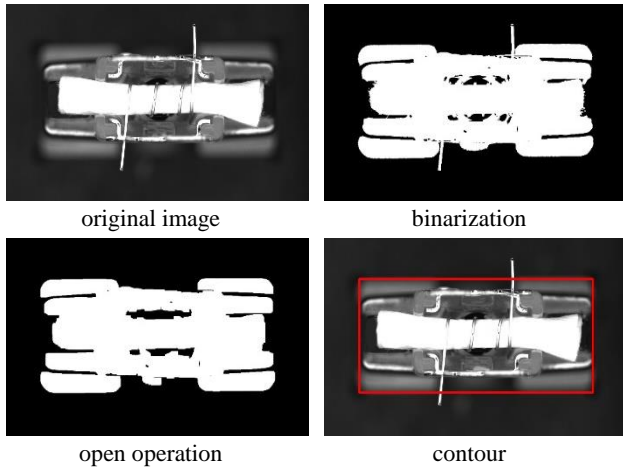
open operation   contour

Figure 2: Detection target location

Step-2: Fixture positioning. Since the wire in the assembly will be outside the scope of the fixture, it needs to be handled by applying an open operation to the image to eliminate protruding wires. Open operations include corrosion and expansion, which are used to

eliminate small objects. After the fixture area is obtained, a contour search is used on the image to obtain the outer rectangular outline of the fixture. The positioning process of the detection target is shown in Figure 2.

Step-3: The corresponding ROI is set according to the occurrence area of various defects. The position of the ROI can be determined based on the relative position of the assembly to the fixture. The ROI settings are shown in Fig. 3. The detection ROI corresponding to different defects is shown in Table 1.

Table 1. ROI Settings

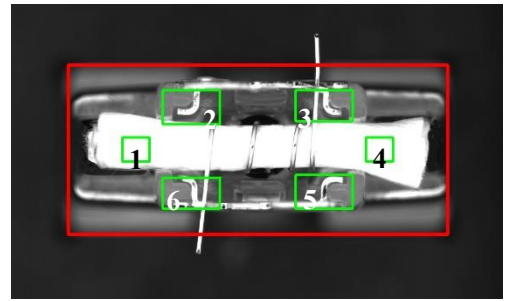| Defect	ROI
Missing cotton core	1, 4
Missing metal sheet	2, 5
Abnormal wire position	4, 6

Figure 3: ROI settings

2.2 Detection of Missing Workpiece

For the detection of missing workpiece defect, the contour filtering method is proposed. Since only the fixture remains after the workpiece is missing, multiple contours appear instead of a full contour when performing a contour search. Contour filtering is to filter the contour of a small area by setting the contour area threshold. The result of the contour filtering is used to judge whether the workpiece is missing. If there is no contour after contour filtering, it can be determined that the workpiece is missing. The contour search result of the sample with missing workpiece is shown in Figure 4.
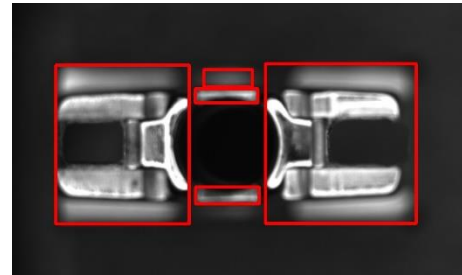

Figure 4: A sample with missing workpiece

2.3 Detection of Missing Cotton Core

For the detection of missing cotton core defect, a pixel statistical method is proposed. The cotton core is bright white when imaged, and the cotton core is judged to be missing by counting the

proportion of white pixels in the ROI. If the ratio is less than the set threshold, it is determined that the cotton core is missing, and vice versa. The steps for white pixel statistics are as follows.

Step 1: Determine whether the pixel is a white pixel. The binarization method is used. A pixel having a gray value greater than 250 is a white pixel. The formula is as follows:

$$g(x,y) = \begin{cases} 1 & f(x,y) > 250 \\ 0 & f(x,y) \leq 250 \end{cases}$$

Step 2: Calculate the proportion of white pixels. First, calculate the number of white pixels, which is equal to the sum of each pixel value. Next, calculate the number of pixels in the ROI, which is equal to the number of columns multiplied by the number of rows. Finally, calculate the proportion of white pixels by the following formula:

$$p = \frac{1}{C \times R} \sum g(x,y)$$

Where p represents the proportion of white pixels, C represents the columns of ROI, R represents the rows of ROI.

2.4 Detection of Missing Metal Sheet

For the detection of missing metal sheet defect, the template matching method is used. The metal claw is a sign of the existence of metal sheet, so that the detection of the metal sheet can be converted into the detection of the metal claw. In order to eliminate the interference of some unrelated regions, we do not directly perform template matching on the original image, but perform template matching on the preprocessed image. Preprocessing operations include closed operations and binarization. The closed operation eliminates black holes, and the binarization operation splits the bright white area, that is, the detected target. Then use template matching for detection. The template matching uses the normalized squared difference method, for which the formula is as follows:

$$R(x,y) = \frac{\sum_{x',y'} \left(T(x',y') - I(x+x',y+y') \right)^2}{\sqrt{\sum_{x',y'} T(x',y')^2} \sqrt{\sum_{x',y'} I(x+x',y+y')^2}}$$

Where $R(x,y)$ represents the result of template matching at (x, y), T represents the template image, I represents the image to be detected. A small value of R means that the difference between the two images is small. We therefore set a threshold of R to determine whether there is a metal claw in the ROI.

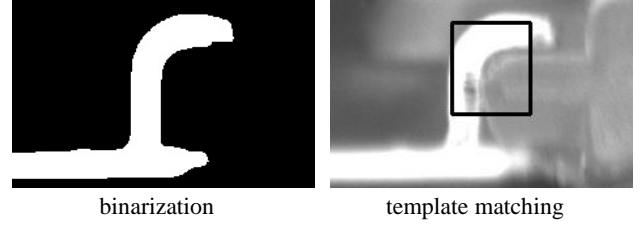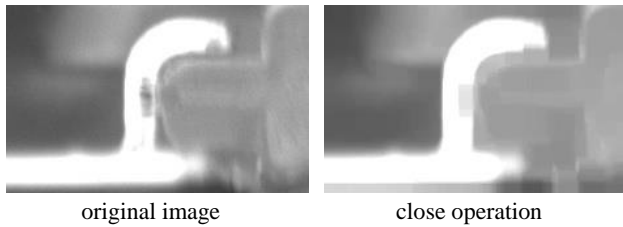The detection process of the sample with the metal piece is shown in Figure 5.

original image   close operation

binarization   template matching

Figure 5: Detection of missing metal sheet

2.5 Detection of Abnormal Wire Position

For the detection of abnormal wire position defect, a measurement distance method based on pixel statistics is proposed. According to the distance between the wire and the vertical side of the metal claw, we can judge whether the position of the wire is abnormal.

(a) original image

(b) pixel statistics
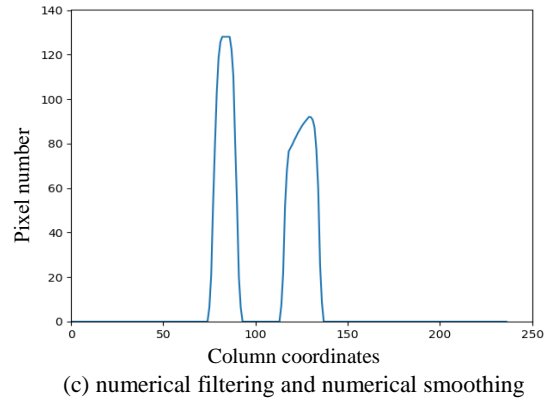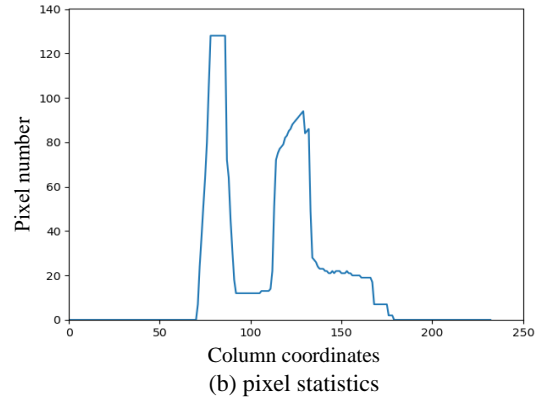
(c) numerical filtering and numerical smoothing

Figure 6: Distance measurement

Similarly, we preprocessed the image, including closing and binarization, to get the target of the detection. Then, the distance between the wire and the metal claw is measured. If the distance is not within the normal range, it is determined that the wire position is abnormal, and vice versa.

The specific steps of the measurement distance method based on pixel statistics are as follows.

Step-1: Calculate the number of white pixels in each column of the ROI, using the pixel statistics method proposed above.

Step-2: Numerical filtering. Set a threshold to filter values less than this threshold in order to eliminate non-detected areas.

Step-3: Numerical smoothing. Smoothing the sequence of values makes it easier to get to the maximum.

Step-4: Calculate the distance. A large number of white pixels appear at the wire and metal jaws, which are two maxima in the numerical sequence. The distance between these two extremes is the distance between the wire and the metal claw.

The process of measuring the distance is shown in Figure 6.

3 Deep Learning Algorithm

3.1 Data Enhancement

Since we don't have sufficient pictures for the training of deep learning algorithms, it is necessary to carry out appropriate data enhancement to increase the diversity and quantity of samples, and thus improve the robustness of the algorithm. According to the characteristics of the detection environment, the following two data enhancement methods are used.

(1) Position offset. The relative position of the fixture and camera can be slightly offset due to mounting accuracy. Randomly shifting the image slightly to improve the adaptability of the algorithm to the installation location.

(2) Brightness and contrast conversion. Different production environments, light sources, and machine vibrations can cause changes in the brightness and contrast of an image. Appropriate brightness and contrast transform enhancements to the image allow the algorithm to adapt to changes in brightness and contrast. The brightness conversion diagram of the image is shown in Figure 7.
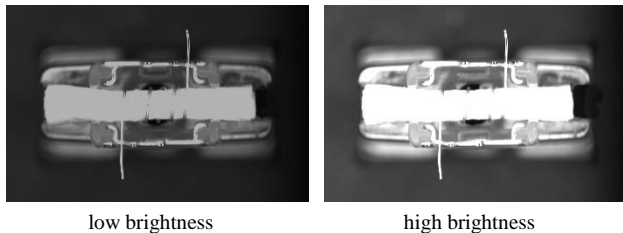

low brightness   high brightness

Figure 7: Brightness transformation

3.2 Model Selection

We made a preliminary selection of the model. We selected several commonly used CNN for experiments, including Alex [6], VGG [7], and MobileNet [8]. We used these networks to build three deep learning models. Then, these models were trained and evaluated. The detection accuracy of each model and the detection time of a single picture are shown as Table 2.

Table 2. Accuracy And Detection Time Of Models

Model	Accuracy	Detection time
Alex	92.61%	24.78ms
VGG	63.04%	105.30ms
MobileNet	97.83%	5.31ms

From the table, it can be found that the MobileNet network has the highest accuracy and the shortest detection time. Therefore, we choose MobileNet for deep learning algorithm design.

MobileNet is a network based on deep separable convolution. The diagram of depth separable convolution is shown in Figure 8. Depthwise separable convolution consist of two layers: depthwise convolution and pointwise convolution. Depthwise convolution applies k×k×1 filter to per each input channel individually. Pointwise convolution applies 1×1×N convolution to create a linear combination of the output of the depthwise layer.

input   depthwise   pointwise   output
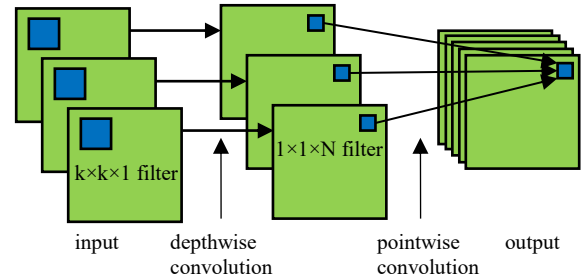   convolution   convolution

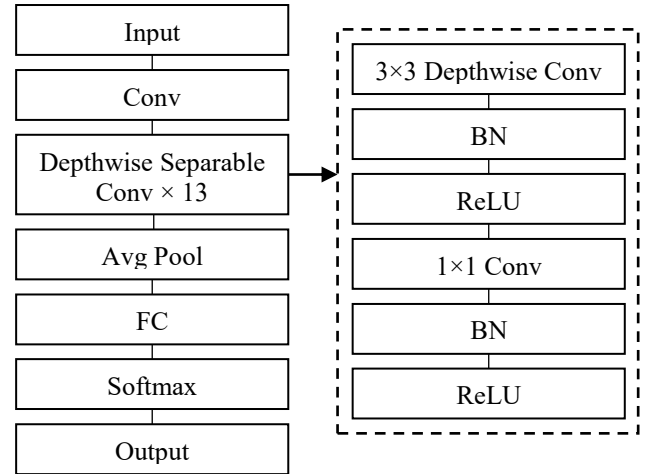**Figure 8: Depthwise separable convolution**

Figure 9: Structure of MobileNet

Compared with standard convolution, the parameter quantity and computation of the depth separable convolution are greatly reduced, which reduces the complexity of the model and improves the detection speed. Therefore, it is more suitable for real-time industrial testing.

MobileNet uses Batchnorm [9] and ReLU nonlinearities after the convolution layer. The overall structure of MobileNet is shown in Figure 9.

3.3 Training Program Improvement

Model training requires continuous traversal of samples. Each sample is a single image. For defect detection tasks, we believe that this training program is not the best. Usually, whether a product is defective or not is based on the difference between it and the normal product. That is, a normal picture is required for defect detection.
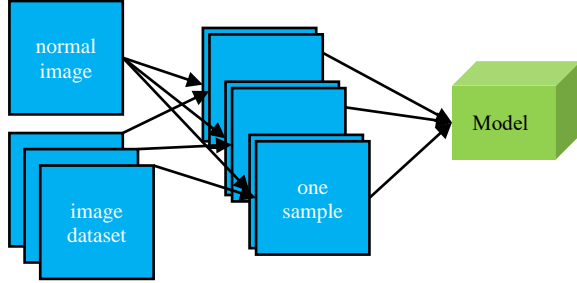

Figure 10: Training program

Zagoruyko and Komodakis [10] used CNN to learn the similarities between the two images and achieved good results. Based on this idea, this paper proposes a new training program. Each sample used in model training consists of two pictures, one is a standard normal picture and the other is any picture in the training set. The diagram of the training program is shown in Figure 10.

4 Experiment And Comparison

4.1 Detection Using Image Processing Algorithm

We used an image processing algorithm for defect detection. Since the detection result of the image processing algorithm largely depends on the setting of the parameters, the principle of setting the parameters of this paper is to make the error rate smaller when the defect detection rate is increased as much as possible. The optimal detection results of the algorithm after constant adjustment of parameters, is shown in Table 3. The detection rate refers to the proportion of samples in this category that are correctly detected. The error rate refers to the proportion of samples that are not in this category, detected as this category.

Table 3. Detection Result Of Image Processing Algorithm

Defect	Detection rate	Error rate
Missing workpiece	100%	0%
Missing cotton core	100%	0%
Missing metal sheet	100%	0.43%
Abnormal wire position	100%	0.85%
Normal	98.72%	0%

From the table, we can see that the algorithm can detect defects 100%. This is because the parameters we set are more stringent. The rate of false positives is not very high and can meet the requirements of detection.

4.2 Detection Using Deep Learning Algorithm

Our deep learning algorithm is designed based on the MobileNet network. We used the original training program and the improved training program proposed in this paper to carry out model training and perform defect detection. The detection results of the original training program are shown in Table 4. The test results of the improved training program are shown in Table 5.

Table 4. Detection Result Of Original Deep Learning Algorithm

Defect	Detection rate	Error rate
Missing workpiece	100%	0%
Missing cotton core	100%	0%
Missing metal sheet	100%	1.41%
Abnormal wire position	83.33%	0%
Normal	97.93%	2.35%

From this table, we can see that the detection rate of the abnormal wire defect has dropped a lot, compared to the image processing algorithm. By observing the images of the abnormal wire defect, we find that the abnormal wire defect is less obvious than other defects, which means that the difference between the image of this defect and the normal image is relatively small. This also shows that the CNN is not sensitive to small changes in the image.

Table 5. Detection Result Of Proposed Deep Learning Algorithm

Defect	Detection rate	Error rate
Missing workpiece	100%	0%
Missing cotton core	100%	0%
Missing metal sheet	100%	0%
Abnormal wire position	83.33%	0%
Normal	100%	2.35%

From this table, we can see that the detection accuracy of proposed training program is improved compared to original training program, but the accuracy of the abnormal wire defect has not been improved.

4.3 Comparison of Tow Algorithms

From the above experiment, we can find the characteristics of the two algorithms. The image processing algorithm has a high defect detection rate and can accurately detect small defects. However, the detection algorithms for each type of defect are different, which makes the algorithm not universal. The deep learning algorithm based on CNN has a high normal detection rate, but the detection rate of small defects is low. However, the deep learning algorithm can be used for the detection of different defects, so the universality is better.

5 Conclusion

This paper proposes two algorithms to detect the assembly defects of the atomizer. We designed a corresponding image processing detection algorithm for different assembly defects, which has a 100% defect detection rate but a lower normal detection rate. We also proposed a deep learning detection algorithm based on the

MobileNet network and our training program, which has a 100% normal detection rate but a lower defect detection rate. Both algorithms have their own advantages and disadvantages. Therefore, when selecting an algorithm for a task the characteristics of the algorithm and the actual requirements should match each other. For industrial detection, it is usually required that the defect can be detected 100%. So, the image processing algorithm is more suitable for our task. In future research, we will try to combine the advantages of the two algorithms to design a new algorithm. Improve algorithm universality while ensuring that defects are correctly detected.

ACKNOWLEDGMENTS

REFERENCES

[1] Jiancheng Jia. A machine vision application for industrial assembly inspection. International Conference on Machine Vision, pp. 172-176, 2010.

[2] Jing Wang, Xiaoyi Yang. Auto-detect of machine vision and its application in assembling inspection. World Congress on Intelligent Control and Automation, pp. 18-22, 2011.

[3] Faisal Ardhy, Farkhad Ihsan Hariadi. Development of SBC based machine-vision system for PCB board assembly automatic optical inspection. International Symposium on Electronics and Smart Devices, pp. 386-393, November 2016.

[4] Je-Kang Park, Bae-Keun Kwon, Jun-Hyub Park, and Dong-Joong Kang. Machine learning-based imaging system for surface defect inspection. International Journal of Precision Engineering and Manufacturing-Green Technology, 3(3):303-310, 2016.

[5] Wu Tong, Chen Ping. X-ray based assembly correctness detection of internal parts of complex structural parts. Laser & Optoelectronics Progress, 55(4):174-182, 2018.

[6] A. Krizhevsky, I. Sutskever and G. Hinton. Imagenet classification with deep convolutional neural networks. Advances in Neural Information Processing Systems, 25:1106–1114, 2012.

[7] Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. arXiv: 1409.1556, 2014.

[8] Howard A G, Zhu M, Chen B, et al. Mobilenets: Efficient convolutional neural networks for mobile vision applications. arXiv: 1704.04861, 2017.

[9] Sergey Ioffe, Christian Szegedy. Batch Normalization: Accelerating deep network training by reducing internal covariate shift. arXiv: 1502.03167v3, 2015.

[10] Sergey Zagoruyko, Nikos Komodakis. Learning to compare image patches via convolutional neural networks. IEEE Conference on Computer Vision and Pattern Recognition, pp. 4353-4361, 2015.