Conductive Particles Detection in the TFT-LCD Manufacturing Process with U-ResNet

Kangping Chen(✉) and Eryun Liu

Department of Information Science and Electronic Engineering,
Zhejiang University, Hangzhou, China
{chenkp,eryunliu}@zju.edu.cn

Abstract. The inspection of conductive particles after Anisotropic Conductive Film (ACF) bonding is a common and crucial step in the TFT-LCD manufacturing process since quality of conductive particles is an indicator of ACF bonding quality. Manual inspection under microscope is a time consuming and tedious work. There is a demand in industry for automatic conductive particle inspection system. The challenge of automatic conductive particle quality inspection is the complex background noise and diversified particle appearance, including shape, size, clustering and overlapping etc. As a result, there lacks effective automatic detection method to handle all the complex particle patterns. In this paper, we propose a U-shaped deep residual neural network (U-ResNet), which can learn features of particle from massive labeled data. The experimental results show that the proposed method achieves high accuracy and recall rate, which exceedingly outperforms the previous work.

Keywords: Conductive particles · TFT-LCD
Deep convolutional network · U-ResNet

1 Introduction

The Anisotropic Conductive Film (ACF) bonding technique has been widely used in the Thin Film Transistor Liquid Crystal Display (TFT-LCD) industry, such as Film on Glass (FOG) and Chip on Glass (COG) [5,12,14]. The ACF is an electrical conductive adhesive, containing small conductive particles distributed on the insulated pad. During the bonding process, the particles deform and make electrical interconnections between the conductive areas on LCD panel and the flexible circuit vertically [14]. In general, validly deformed particles have clear bright parts and dark parts to be seen when projecting tilted light (see Fig. 1). As the bonding conditions are critical, such as pressure, temperature, time and alignment of the parts, the particles may fail in deformation and be unable to make electrical conductions if any of them is not satisfied (see Fig. 2) [6]. Insufficient number of valid particles may result in poor conductivity and even electrical failure between the panel and flexible circuit [6,14], which leads

J.-H. Lai et al. (Eds.): PRCV 2018, LNCS 11259, pp. 162–173, 2018.
https://doi.org/10.1007/978-3-030-03341-5_14

to inferior products and waste of material. To guarantee the electrical quality at the bonding step and not affect the following manufacturing procedures, an after-bonding inspection of conductive particles is indispensable.

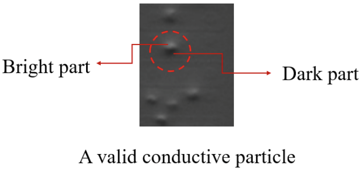

Fig. 1. An example of valid particle, which has a bright part and a dark part to be seen when projecting light.

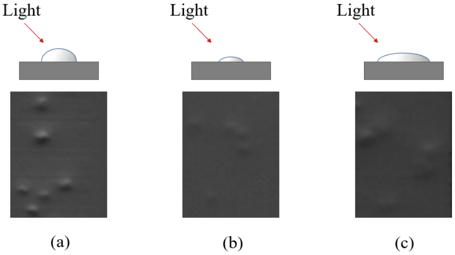

Fig. 2. **(a)** Valid conductive particles; **(b)** invalid particles with too little deformation; **(c)** invalid particles with too much deformation.

The current practice of after-bonding inspection is to detect the number of valid conductive particles under the microscope manually. Fully automatic particle inspection is still an open problem due to numerous factors, such as clustering and overlapping of the conductive particles after the bonding process (see Fig. 3). We see that sometimes it is even hard to distinguish the particles on such small area, not to mention detecting the valids.

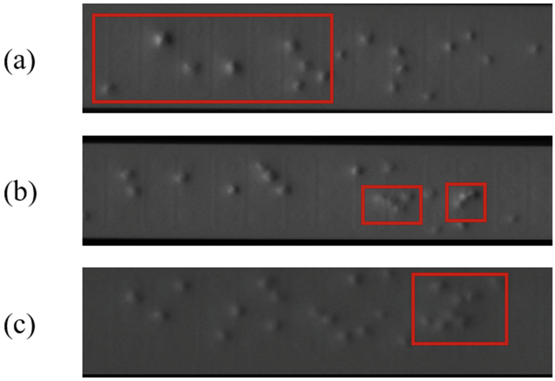

Fig. 3. Challenges in the particle detection. **(a)** *Variant size*: The particles in the box are valid but in different size; **(b)** *overlapping and mixture*: The particles in the box overlaps while the valids and invalids are in mixture; **(c)** *poor illumination and clustering*: The box region is much darker than it in **(a)** or **(b)**. Meanwhile, the particles are in clusters.

With the development of Automatic Optic Inspection (AOI) technology, a series of researches [5,6,9,12,15] have been made to address the problem. These researches focus on how to design appropriate description to the valid particles. In Lin's (2011) [6], the particles are extracted with *Prewitt operator*, utilizing the gradient feature around them. After processing the image with Prewitt operator, the Otsu binarization is carried out on the processed image. Then on the binarized image, template matching is used to further localize valid particles. However, Prewitt operator performs poor when particles overlap and is sensitive to noise. Moreover, Otsu thresholding is based on the hypothesis that the image can be binarized by a global grayscale threshold [11], which is not practical in the complex environment of circuit surface. Later in Chen's (2017) [15], the particles are extracted by the *intensity difference*. Observing the drawbacks of Lin's [6], the authors substitute background subtraction method for the Prewitt operator process to divide the images into background and foreground, also suppress the noise. And then, instead of a simple binarization, k-means is used to classify the pixels, which is more suitable for the overlapping situation and $k = 4$ shows the best performance in the authors' experiment. Though the method shows higher precision than Lin's [6], it is more time-consuming and still hard to cope with complex situation. In addition to the pure 2D analyses, Guangming Ni [9] designed a special differential interference contrast (DIC) system to detect the particles in 3D space with predefined parameters. Though it gains more information in the additional dimension, the detection is still done by handcrafted features. The handcrafted features are created by the researchers from limited samples. Since the conductive particles vary in shape and size, these handcrafted-feature based method may fail to deal with complex situations.

In this paper, we incorporate the learning-based idea [2,7] and design a convolutional network for valid conductive particles detection. The proposed network architecture is inspired by U-NET [10], which has been widely used in medical analysis [1,10] and autonomous driving [13]. Based on the U-shaped network, we add skip connections to cascade convolutional layers to make residual blocks [4], which is helpful to recover the full spatial resolution at the network output [1]. Also, instead of solving the detection problem in a classification manner, our task is reduced to a more straight-forward regression problem. Besides, we make a series of improvements for better fitting our task. The experimental results show our method outperforms the previous methods both in precision and recall rate.

2 Proposed AOI System for Particle Detection

2.1 Overview

The AOI system is designed to work on the assembly line. As shown in Fig. 4, the full pipeline of our AOI system includes three major steps:

1. Image acquisition and ROI extraction. The raw image is obtained by a line scan camera. One image includes thousands of pads, each of them is a ROI containing particles. The ROIs are detected by aligning the marker points which located on the corner of LCD panel.

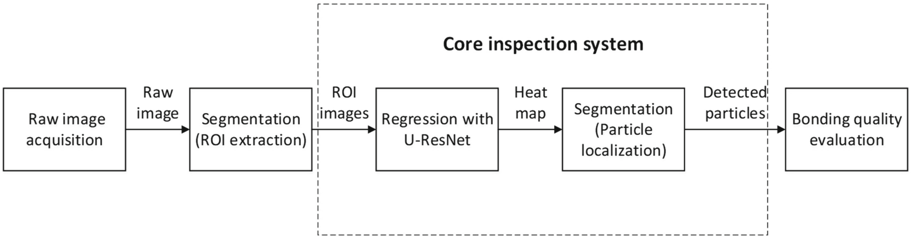

Fig. 4. The full pipeline of our proposed AOI system.

2. Particle detection. This step is the focus of our work in this paper. Given a list of ROI images, the proposed algorithm detects valid particles on the ROIs one by one.
3. Bonding quality evaluation. With the detected particles on each pad, a final decision about the effectiveness of ACF bonding is made based on the quality of particles, such as number of valid particles.

2.2 Pixelwise Regression

As shown in the flowchart (see Fig. 4), the particle detection is separated into two problems, namely particle heat map estimation and particle localization.

The particle heat map is a map with value range in $[0, 1]$ indicating strength of particle. A higher value indicates higher probability that there is a particle. The heat map estimation can be modeled as a pixelwise regression problem as follows:

$$\mathbf{Y}^* = F(\mathbf{X}), \tag{1}$$

where $\mathbf{X}$ is the input ROI image, $\mathbf{Y}^*$ is the output heat map of the same size as $\mathbf{X}$, and F is the pixelwise regression model, such as deep neural network, that mapping the ROI image to the heat map image.

2.3 U-ResNet: U-Shaped Architecture with Residual Blocks

In general, there are two types of technologies for particle strength heat map estimation, namely the knowledge driven and the data driven methods.

The knowledge driven based methods estimate the heat map by extracting handcrafted particle features, while the data driven methods learn the estimation model from markup data.

The existing methods are mainly based on the handcrafted features. However, there are rare methods based on learning from data in the literature. Learning based methods have the advantage to handle complex background.

In this section, we proposed a data driven method based on deep learning. The network structure is shown in Fig. 5.

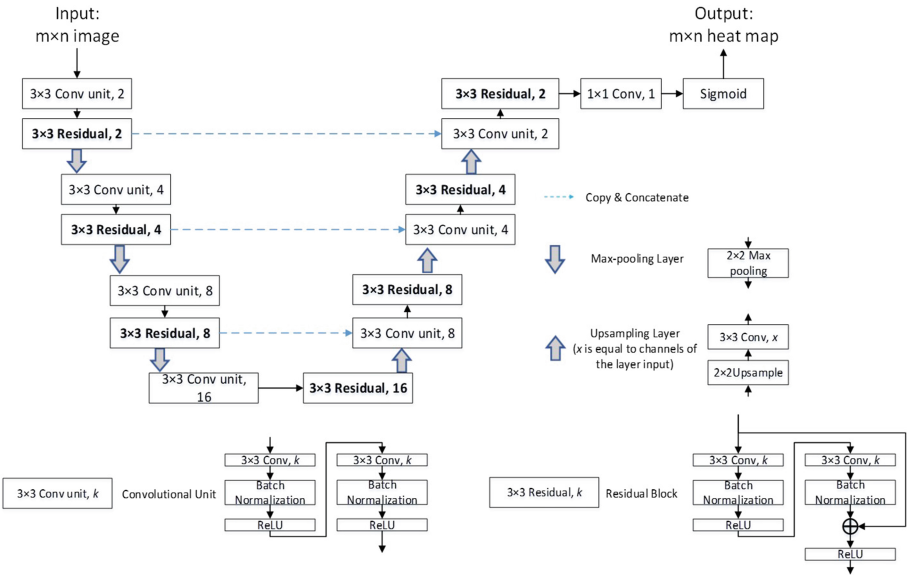

Fig. 5. The architecture of U-ResNet-7.

Architecture. The proposed U-ResNet is a U-shaped network with a down path and an up path, consisting of 14 basic blocks, a 1 × 1 convolutional layer and a sigmoid layer as output, as shown in Fig. 5. The basic block refers to the convolutional unit or the residual block, either composing of two 3 × 3 convolutional layers with stride 1 and zero-padding, each followed by a batch normalization layer and a rectified linear unit (ReLU). Differently, the residual block contains additional shortcut connection between the layer input and the latter ReLU. Every two blocks makes a step. After taking a step, a 2 × 2 max-pooling layer is followed in the down path while an upsampling layer is in the up path.

It should be noted that the proposed network architecture is inspired by the U-NET [10] and ResNet [4]. The feature channels are doubled every step on the down path and halved on the up path, which is brought from U-NET [10]. The structure of *copy & concatenate* is also adopted for combining the feature hierarchy and refining the spatial precision [8].

The network shown in Fig. 5 includes 7 residual blocks, which we call U-ResNet-7. This network structure can be further extended to have deeper layers. By substituting residual blocks for the remaining convolutional units, the U-ResNet-14 is built. By further replacing the three convolutions in the upsampling layers, we design the U-ResNet-17, which doesn't contain any seperate convolutional layer except for the last 1 × 1 convolution.

Heat Map Regression. The output layer in our networks is a sigmoid layer, which normalizes the output to range $[0, 1]$. Overall, the network takes an $m \times n$

image as input and generates a heat map of the same size. Each value on the heat map indicates the strength of being a valid conductive particle center at the corresponding position on the ROI image.

2.4 Loss Function

Given the predicted heat map and label image, the loss function defined for the regression task is

$$L(\mathbf{Y}^*, \mathbf{Y}) = \sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n} \alpha_{ij}(y_{ij}^* - y_{ij})^2}, \tag{2}$$

where m and n refer to the size of the input image, $\mathbf{Y}$ represents the label map of the conductive particles while $\mathbf{Y}^*$ refers to the predicted heat map, y_{ij} and y_{ij}^* stands for the strength value at (i, j) on the respective maps. As the number of pixels of particles and background in the label image is unbalanced, the weight coefficient α_{ij} is large for the particles and small for the background, which penalizes more on the labeled particles while keeps the robustness for the wrong labels, as shown in Eq. (3).

$$\alpha_{ij} = 0.05 \times (1 - y_{ij}) + 0.95 \times y_{ij} = \begin{cases} 0.05 & \text{if } y_{ij} = 0, \\ 0.95 & \text{if } y_{ij} = 1. \end{cases} \tag{3}$$

2.5 Particle Localization

Given the predicted heat map, the particle can be localized in two steps:

1. Particle segmentation. The value in the heat map indicates the strength of being center of particle. A threshold k can be selected to segment the particle region.
2. Given the segmented particle area, the particle center is estimated as the centroid of segmented objects.

3 Experiments

3.1 The Dataset

As far as we know, there is no conductive particle dataset in the public domain that is available for research. A large amount of data is necessary for both training and validation. We created a dataset for algorithm evaluation. The training and validation datasets are both created in two steps.

1. ROI Extraction: The raw images are of 8-bit grayscale, captured by a 64 kHz, 4k TDI Line Scan camera of 0.72 μm/pixel in resolution. The size of raw image is 42036 × 1635. As the particles only appear on the pin area of LCD screen,

Table 1. Detailed information of the annotated datasets.

Dataset Name	Set-A	Set-B	Set-C
Image Size	1024 × 128	128 × 64	92 × 24
Training Set Count	3304	5665	5792
Validation Set Count	827	1417	1448
Total Count	4131	7082	7235

the detection algorithm focuses only on the ROI images. There are three different sizes of ROI images according to the pin size of LCD screen used tested in our experiments, namely 1024×128, 128×64, and 92×24, which we call Set-A, Set-B and Set-C, respectively. The particle characteristics on these three types of ROI images are slightly different.

2. Particle Location Annotation: Given a ROI image $\mathbf{X}$, the label image $\mathbf{Y}$ of the same size is obtained, where $y_{ij} = 1$ with (i, j) is the manually marked center of particles otherwise $y_{ij} = 0$. The ROI images along with corresponding label images in three categories compose the final datasets called Set-A, Set-B and Set-C (see Table 1).

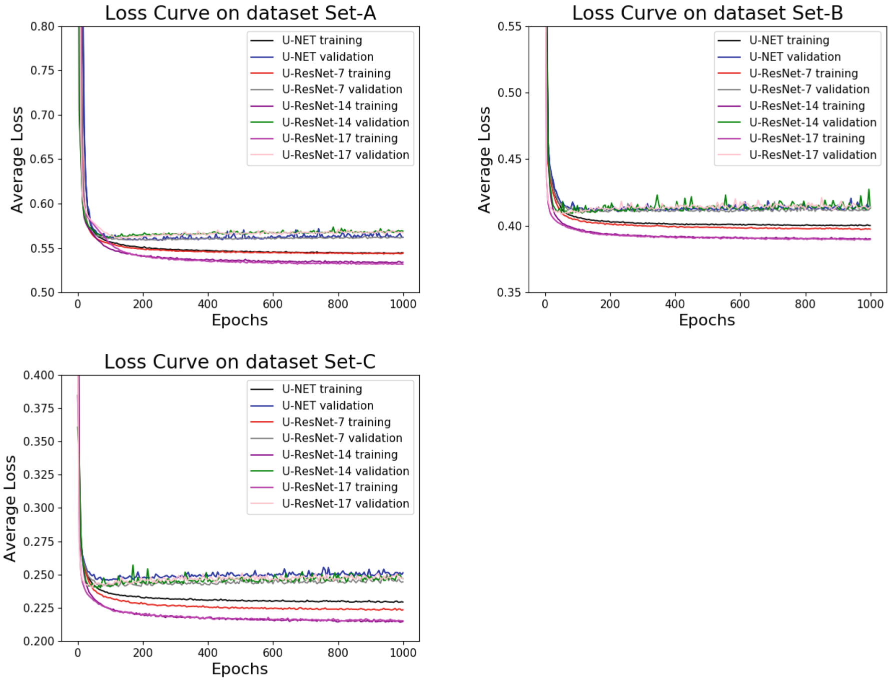

Fig. 6. The loss curves on three datasets.

3.2 Training the Model

To speed the training process, we divide the grayscaled value of pixels by 255 to normalize the values into range $[0, 1]$. We also added a batch normalization layer after each convolutional layer (see Fig. 5).

The weights of our network are first initialized by the way introduced by Kaiming He [3]. Then the model is trained via backpropagation algorithm with GPU accelaration. We use Adam optimizer with a batch size of 64 samples, learning rate of 0.001, the first momentum $\beta_1 = 0.9$, the second momentum $\beta_2 = 0.999$, and weight decay of 0.01. It takes about 20 h to train for 1000 epochs on a single NVIDIA TITAN X GPU. The training and validation loss curves are shown in Fig. 6. As we can see in the figure, both loss curves are converged.

From the loss curves, we see that all U-ResNet based networks have lower loss than the original U-NET [10]. The U-ResNet-14 and U-ResNet-17 have the lowerest training loss. But on the validation sets, U-ResNet-7 always has the lowerest loss, which indicates that U-ResNet-7 may have the best particle localization performance.

3.3 Precision and Recall Evaluation

Previous loss analysis indicates that U-ResNet-7 has the best convergence performance on dataset Set-A while U-ResNet-14 wins on Set-B and Set-C. But the output from the networks is actually a heat map rather than the detection results. As mentioned above, we need to determine the segmentation threshold k for further processing to localize particles. To evaluate the final results, we introduce precision and recall as quantitative criterions.

The precision and recall are defined as:

$$Precision = \frac{|U \bigcap U^*|}{|U|} \times 100\% \tag{4}$$

$$Recall = \frac{|U \bigcap U^*|}{|U^*|} \times 100\% \tag{5}$$

where U is the set of ground truth particles on the label map, U^* is the set of localized particles, the intersection $U \bigcap U^*$ refers to the set of correctly localized particles, and $|.|$ is the number of elements in the set. To determine the intersection, the location (i^*, j^*) of each particle in set U^* and the location (i, j) of its nearest neighbor in set U is measured with Euclidean distance D, as shown in Eq. 6.

$$D = \sqrt{(i^* - i)^2 + (j^* - j)^2}. \tag{6}$$

The tolerance of D is set to 5 in the experiment, which corresponds to the nearly maximum radius of valid particle in our dataset.

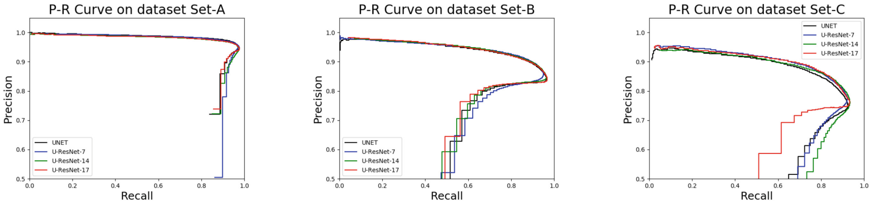

Fig. 7. Precision-Recall curves of the proposed model.

Table 2. Performance comparison on three datasets.

		Lin's [6]	Chen's [15]	U-NET [10]	U-ResNet-7	U-ResNet-14	U-ResNet-17
Set-A	Precision	74.8%	77.5%	96.0%	**96.1**%	95.9%	96.0%
	Recall	74.8%	77.5%	96.0%	**96.1**%	95.9%	96.0%
Set-B	Precision	83.8%	81.0%	89.6%	**89.9**%	89.8%	89.7%
	Recall	31.0%	33.1%	89.6%	**89.9**%	89.8%	89.7%
Set-C	Precision	75.4%	77.0%	83.5%	84.5%	**84.7%**	84.6%
	Recall	30.9%	30.7%	83.5%	84.5%	**84.7%**	84.6%

Tradeoff Between Precision and Recall. To evaluate the tradeoff between precision and recall, the P-R curves with respect to threshold k on heat map are shown in Fig. 7.

We see that all 4 networks have similar performance on dataset Set-A. But on Set-B and Set-C, whose image data contains denser particles with more variant size, we observe that the curves are discriminate. The curve of U-ResNet-17 decays the most saliently after reaching the turning point, especially on Set-C. Combined with loss curve of U-ResNet-17, we infer that it's caused by slightly overfitting, because U-ResNet-17 with the most layers among the models, has the best training error but performs poor on the validation set. The curve of U-NET [10] decays the second most saliently while the U-ResNet-7 and U-ResNet-14 perform excellent on three datasets.

The curves also manifest that we cannot make both precision and recall close to 1. To tradeoff between the two criterions, we seek for a threshold k that makes them close to each other. The equal value determines the best precision and recall on the dataset (see Table 2). In accord with the result from loss curve, U-ResNet-7 has the best precision and recall on dataset Set-A and Set-B while U-ResNet-14 performs the best on Set-C. U-ResNet-7 and U-ResNet-14 outperforms U-NET [10] on all datasets, which demonstrates that U-ResNet is more suitable for our particle detection task.

3.4 Comparison with Traditional Methods

The biggest difference between the traditional methods and the proposed method is how to extract features. Figure 8 shows several examples from three datasets. As we can see, the method based on U-ResNet or U-NET [10] performs better than the traditional methods. Lin's method [6] has poor performance in region with dense or overlapping particles, especially on Set-B and Set-C. Chen's [15] has better result than Lin's method but still loses a number of particles.

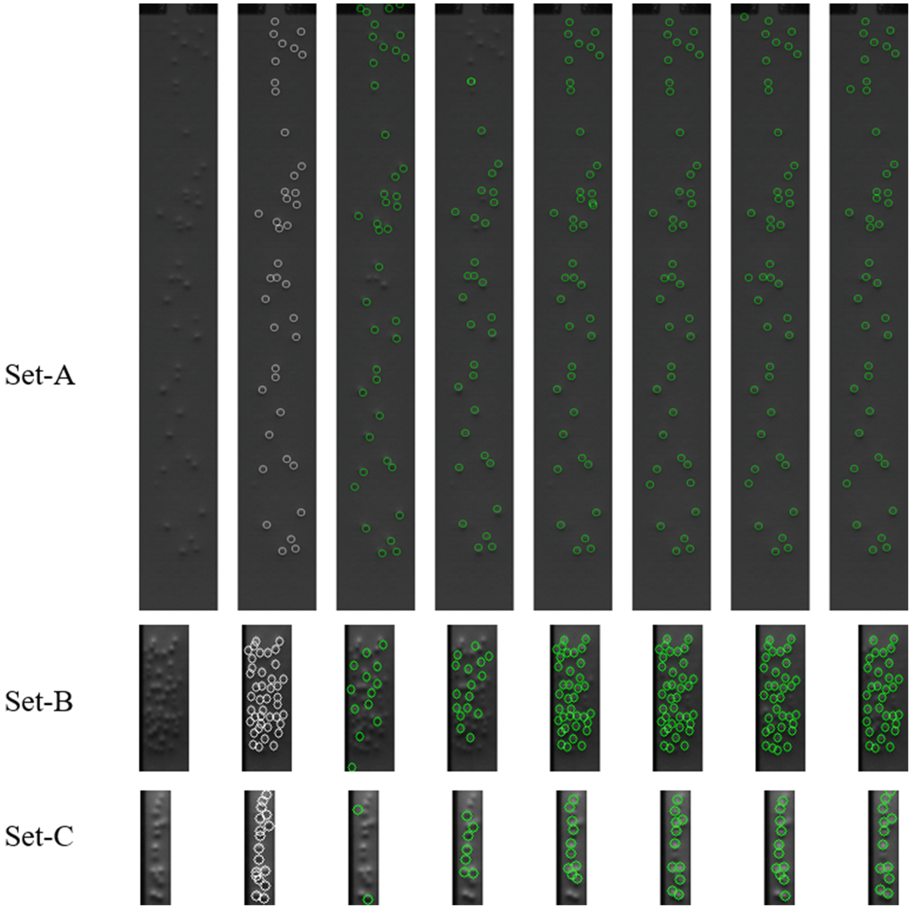

Fig. 8. Examples of particle detection results on three datasets. From left to right: ROI image, Ground truth, Lin's [6], Chen's [15], U-NET [10], U-ResNet-7, U-ResNet-14, U-ResNet-17

Table 2 illustrates the precision and recall values of the methods on three datasets. We see on the table that on Set-B and Set-C, the two traditional methods have very low recall rate while all deep learning based methods exceedingly outperform the two traditional methods. This fact implies that learning-from-data methods are more suitable for detecting valid conductive particles.

3.5 Computation Time

The system was implemented on a Ubuntu 14.04 LTS OS system with 3.5 GHz i7-5930 CPU, 32 G RAM and TITAN X GPU. The total computation time comparison is illustrated on Table 3. In our experiment, the regression stage accounts for less than 1% time cost with GPU acceleration, while more than 99% of time is consumed for localizing particles on the heat map. We see the method based on U-ResNet or U-NET performs much better than the traditional ones and U-ResNet-7 is more time-saving.

Table 3. Average computation time comparison (unit: ms).[1]

	Lin's [6]	Chen's [15]	U-NET [10]	U-ResNet-7	U-ResNet-14	U-ResNet-17
Set-A	11.08	151.11	**5.37**	5.44	6.60	7.21
Set-B	9.28	21.92	2.61	**2.58**	2.64	2.66
Set-C	8.45	14.10	1.98	**1.96**	2.05	2.06

[1] For Lin's [6] and Chen's [15] methods, no GPU was used in the evaluation, while for the deep learning based methods, the regression step was tested with GPU acceleration.

4 Conclusion

The detection of conductive particles is common and crucial for the TFT-LCD manufacturing process. Due to the complex pattern of conductive particles, there has not been a perfect solution to detect them with certain handcrafted features. In this paper, we make two major contributions. First, we apply deep convolutional network to extracting features of conductive particles. Based on the prevailing architecture of U-NET and residual blocks, the U-ResNet architecture is proposed for better fitting our task. Second, we transform the particle detection into a pixelwise regression problem. Then valid conductive particles can be directly detected from the heat map.

References

1. Drozdzal, M., Vorontsov, E., Chartrand, G., Kadoury, S., Pal, C.: The importance of skip connections in biomedical image segmentation. In: Carneiro, G., et al. (eds.) LABELS/DLMIA -2016. LNCS, vol. 10008, pp. 179–187. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46976-8_19
2. Ferguson, M., Ak, R., Lee, Y.T.T., Law, K.H.: Automatic localization of casting defects with convolutional neural networks. In: 2017 IEEE International Conference on Big Data (Big Data), pp. 1726–1735. IEEE (2017)
3. He, K., Zhang, X., Ren, S., Sun, J.: Delving deep into rectifiers: surpassing human-level performance on imagenet classification. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 1026–1034 (2015)

4. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778 (2016)
5. Jia, L., Sheng, X., Xiong, Z., Wang, Z., Ding, H.: Particle on bump (POB) technique for ultra-fine pitch chip on glass (COG) applications by conductive particles and adhesives. Microelectron. Reliab. **54**(4), 825–832 (2014)
6. Lin, C.S., Huang, K.H., Lin, T.C., Shei, H.J., Tien, C.L.: An automatic inspection method for the fracture conditions of anisotropic conductive film in the TFT-LCD assembly process. Int. J. Optomechatronics **5**(3), 286–298 (2011)
7. Lin, H., Li, B., Wang, X., Shu, Y., Niu, S.: Automated defect inspection of led chip using deep convolutional neural network. J. Intell. Manuf., 1–10 (2018)
8. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3431–3440 (2015)
9. Ni, G., Liu, L., Du, X., Zhang, J., Liu, J., Liu, Y.: Accurate AOI inspection of resistance in LCD anisotropic conductive film bonding using differential interference contrast. Opt. Int. J. Light. Electron Opt. **130**, 786–796 (2017)
10. Ronneberger, O., Fischer, P., Brox, T.: U-Net: convolutional networks for biomedical image segmentation. In: Navab, N., Hornegger, J., Wells, W.M., Frangi, A.F. (eds.) MICCAI 2015. LNCS, vol. 9351, pp. 234–241. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-24574-4_28
11. Sezgin, M., Sankur, B.: Survey over image thresholding techniques and quantitative performance evaluation. J. Electron. Imaging **13**(1), 146–166 (2004)
12. Sheng, X., Jia, L., Xiong, Z., Wang, Z., Ding, H.: ACF-COG interconnection conductivity inspection system using conductive area. Microelectron. Reliab. **53**(4), 622–628 (2013)
13. Wirges, S., Hartenbach, F., Stiller, C.: Evidential occupancy grid map augmentation using deep learning. arXiv preprint arXiv:1801.05297 (2018)
14. Yen, Y.W., Lee, C.Y.: ACF particle distribution in COG process. Microelectron. Reliab. **51**(3), 676–684 (2011)
15. Yu-ye, C., Ke, X., Zhen-xiong, G., Jun-jie, H., Chang, L., Song-yan, C.: Detection of conducting particles bonding in the circuit of liquid crystal display. Chin. J. Liq. Cryst. Disp. **32**(7), 553–559 (2017)