Pin of the IC chips recognition

Haiwei Fu
CentraleSupelec
CentraleSupélec, 3 Rue Joliot Curie, 91190 Gif-sur-Yvette
haiwei.fu@student-cs.fr

Hooman Halabichian
CentraleSupelec
CentraleSupélec, 3 Rue Joliot Curie, 91190 Gif-sur-Yvette
hooman.halabichian@student-cs.fr

Guchao Zeng
CentraleSupelec
CentraleSupélec, 3 Rue Joliot Curie, 91190 Gif-sur-Yvette
guchao.zeng@student-cs.fr

Abstract

In industry, chip is a very important link. Because many electronic products now rely on chips, such as computers, mobile phones, and even extremely sophisticated precision instruments. More importantly, how to identify those mass production of various types of chips, can be said to be imminent application. At present, as the most cutting edge such as unmanned autonomous driving chip, robot chip, drone chip, chip and its quality is the most critical technology and link. So the chip issue is too important. If our technology is greatly improved or iterated, chip processing will be more efficient. But for a long time, it is difficult to improve the productions' efficiency with quality of chips, by relying on the manual and process of workers in some assembly lines. Now, if we can use robots to recognize the chip automatically and start the process of automation assembly line, could greatly improve production efficiency and quality. Computer vision in recognition is thought to be an essential part of this. Our project is to be designed to give some ways or improve this problem.

It is good to know that with the development of technology, there comes more and more research and algorithms in solving and increase the efficiency of the industrial problems, and comes a more rapidly iteration of the technologies. Our project is keep up with the latest computer vision methods and algorithms, to exploration a way to detect and recognise pins of IC chips.

1. Introduction

The project is for pins of IC chips recognition using the very latest and commonly popular computer vision methods, such as Gaussian filter, edge detection, Corner detection, Otsu's method SIFT an so on, to at first select the object's features, then analyse, use mainly K-means, k-Nearest Neighbor and confusion matrix to do a data set based recognition, finally give the records and predictions. The goal is to recognise pins of IC chips. As there are often 20 plus types of pins, we select the most common and useful IC chips' pins, which is the 8 + 8 16 pins IC chips to do the project. Also, we did a IC chips with pins and IC chips without pins detection and recognition.

IC chips can be divided into many categories. For the use of Common IC chips package, there are often 21 types of categories, such as Pin Grid Array, FQFP (fine pitch quad flat package), DSO (dual small out-lint) and so on. We need to do the pin recognition within those many chips from different types, and from those different looking pins, select the IC chips with pins or without pins, also select the one classical type 16 pins IC chips and not 16 pins IC chips.

Meanwhile, we would like to know and test that we use those very popular and latest well known computer vision methods and algorithms in IC chips detection and recognition, at which level the methods and algorithms would be suited for this kind of task? What are the performances of those methods and algorithms? What are the accuracy of those methods and algorithms? What are the pros and cons of those methods and algorithms? If we could do a comparison and evaluation? If we could find something, What could be evaluated, or pay more attention, or what could to improve next?

Because computer vision is a cutting edge domain, and using it can dramatically increase productivity, it has a profound and valuable application prospect. Moreover, the domain develop very fast, but not very mature. Even we have to focus on from a small point cut. And as the key link in industry, it is necessary to master the key technology and carry out better technical iteration.

The potential applications like Computer vision, Machine vision, chip recognition, especially the chip recognition with machine vision, it is a very cutting edge domain.

With the rapid development of semiconductor technology and electronic industry, chip defect detection technology plays an increasingly important role in the whole industrial production process.

In the process of chip production and manufacturing, each process flow is interlinked, the technology is complicated, and the slight variation of materials, environment, process parameters and other factors often lead to chip defects and affect product yield.

As a key link in the chip production line, chip quality inspection can actively feedback the product quality information, so that people can timely control the health status of each production link, and promote the quality inspection technology in the production line more and more prominent.

There are three ways to detect the appearance of the chip: Traditional manual detection method; Using laser measurement technology for chip appearance detection; Detection method based on machine vision.

Manual visual detection method is gradually replaced by automatic detection technology because of its shortcomings such as low efficiency, low precision, high cost, high labor intensity and inconsistent standards. Compared with the first two methods, the chip defect detection technology based on machine vision has been widely studied and applied as a flexible, real-time, non-contact and high-precision detection technology.

In recent years, the successful application of deep learning model represented by convolutional neural network in the field of vision also provides a new development direction for defect detection.

There have many chip defects. Chip production needs to go through several processes, among which each process is interlinked, including chip design, manufacturing, packaging and testing four major links.

Especially before packaging, process operation control, process parameters, environment and other factors will have a certain impact on chip quality, resulting in various types of defects, different forms, complex background and other characteristics.

It can be roughly classified as poor raw materials, foreign bodies, scratches, Bump element defects (bumps, misalignments, or missing), gold contaminants, and etching fluid dirty residue. In addition, small sample defects such as etching rust, excessive plating, discoloration and wire damage may occur during the chip manufacturing process.

And also packaging defects include printing defects and pin defects. The printed symbol on the surface of the package indicates its name, specification, model and performance information, which is an important basis for identifying the chip.

Clear symbol is the inherent demand of high quality chip. The printing defects often exist in actual production include wrong character, offset, leakage, multiple printing, fuzzy, tilt, displacement, hyphenation, double-layer printing and no word, etc. Pin defects include missing pins, broken pins and bent pins.

Therefore, the main detection contents of chip appearance defect detection can be summarized into the following three points:

1. Contents of package detection include: scratches, stains, damage, not full, spillover, etc.

2. the content of printing detection includes: wrong character, offset, leakage, multiple printing, fuzzy, tilt, displacement, hyphenation, double-layer printing, no font, etc.

3. Pin detection includes: pin missing, pin damage, pin spacing, pin width, pin bending, pin span, pin length difference, pin standing height, pin coplane, pin tilt, etc.

Moreover, there have many advantages of machine vision application in chip detection.

At present, the integration degree of various semiconductor chips is becoming higher and higher, and the volume of chips tends to miniaturization and miniaturization, which puts forward higher requirements for chip detection.

In the field of industrial automation production, visual appearance detection can be said to be the most core part. No matter from object to bar code identification, or product quality detection and appearance measurement, visual appearance detection technology has played an important role.

2. Problem definition

The project can divide as a recognise problem, to recognise the pins of IC chips. Our first part is to recognise the IC chips with pins or the IC chips without pins. Then second is to choose a typical type the 16 pins IC chips, to recognise the IC chips is 16 pins or not 16 pins IC chips.

So, based on this recognition problem, we think we need to do some notation, but we then think it is not a deep learning based project, we did not do the label progress, but we then use a more computer vision way to do, like the k-mean, the cluster methods, which use in a unsupervised learning, given the images present their distribution and so on, a more statistical way to do the notation. After that test the accuracy and do prediction. We also test the k-Nearest Neighbor, which is a supervised learning method, and assumes that given a training data set, the class of instances it is determined.

3. Related work

The current computer vision chip recognition, more inclined to the combination of machine vision and chip recognition. Mainly chip character recognition, chip edge, corner, image, shape and other features recognition. In terms of research, many research directions on robustness and frame rate of visual feature. Extraction algorithm are proposed for these application scenarios, among classic feature extraction algorithms such as SURF, ORB, HOG and LBP, SURF is relatively robust, but it relies too much on the selection of the main direction, resulting in insufficient robustness of direction change. SIFT algorithm can extract robust local features with invariance from images, and has the strongest robustness to direction change, illumination change, noise, debris scene and occlusion effect, etc., meeting the needs of unmanned driving technology. The problems of low frame rate and high power consumption caused by large amount of calculation of SIFT algorithm can be solved by designing a special hardware accelerator chip with high parallelism.

Improving hardware facilities and their performance is also a key link. There are also many research to do research at its hardware.

4. Methodology

4.1. Dataset and images collection

We collect images the data set from many websites, such as Taobao, Amazon. Because those e-commence are selling many IC chips products and provide a very good quantity of images.

And for the recognition, we have a data set that has train set and test set, the train set has nearly 100 images and the test set has approximately 30 images. using K-means, a unsupervised learning method. Then give the accuracy etc.

Other for the recognition, we test in a way in small sample data set, which only has 22 images. And we divide each images into a classification, and use k-Nearest Neighbor, a supervised learning method. Then give the accuracy etc.

4.2. Image feature selection: on hand algorithms

For image feature selection, we use two different ways to do that. One way is for using algorithms on hand. By using this way, we need to build the models and algorithms by ourselves. The advantages of this on hand build models are that it could allow us to do the edit or modify in the inner algorithms, like a white box. Those algorithms are very classic and commonly used, And like a adjust parameters step by step in the inner algorithms, it is very flexible.

4.3. Image feature selection: opencv algorithms

Another way we use is use the opencv algorithms directly. The advantages are that it is very convenient and very quick. The algorithms are given, we just need to call the many libraries and functions. The effects of the opencv is very good, like the good color, the canny detection the effect is more accuracy than the on hand algorithms, and more robust. But for the use of the Otsu's method, the adjust parameter progress that help us find the best particularly threshold is mandatory, like a black box operation.

4.4. Image feature selection: methods and algorithms

Gaussian filter

Gaussian filter is a linear smoothing filter, suitable for eliminating Gaussian noise, widely used in image processing noise reduction process. Generally speaking, Gaussian filtering is the process of weighted average of the whole image, and the value of each pixel is obtained by weighted average of its own and other pixel values in the neighborhood. The specific operation of Gaussian filtering is to use a template (or convolution, mask) to scan every pixel in the image, and use the weighted average gray value of the pixel in the neighborhood determined by the template to replace the value of the central pixel of the template. Gaussian filtering (Gaussian smoothing) is the most common operation in image processing and computer vision.

It can be simply understood that Gaussian filtering denoising is the weighted average of pixel values in the whole image, and the value of each pixel is obtained by weighted average of its own value and other pixel values in the neighborhood.

The specific operation of Gaussian filtering is to use a user-specified template (or convolution, mask) to scan every pixel in the image, and use the weighted average gray value of the pixel in the neighborhood determined by the template to replace the value of the central pixel of the template.
One-dimensional Gaussian distribution:

$$G(x) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2}{2\sigma^2}}$$

Two-dimensional Gaussian distribution:

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$$

In image processing, Gaussian filtering is generally realized in two ways, one is by discretization window-sliding convolution, the other is by Fourier transform. The most common is the first sliding window implementation. Only when the discretized window is very large and the computation with sliding Windows is very large (that is, the implementation using separable filters), the implementation method based on Fourier change may be considered.

Bilateral filter

Bilateral filtering is a nonlinear filtering method. It is a compromise processing combining spatial proximity and pixel value similarity of the image, while considering spatial and information and gray similarity, so as to achieve the purpose of edge preserving and denoising. It has the characteristics of simple, non-iterative and local processing. The filter effect of edge preserving denoising can be achieved because the filter is composed of two functions: one is determined by geometric space distance and the other is determined by pixel difference.

The two-sided filter can save the edge details of the image well and remove the noise of the low frequency component. However, the efficiency of the two-sided filter is not too high, and the time taken by the two-sided filter is longer than other filters.

For simple filtering, you can set the two sigma values to the same value. If the value is <10, it will have little effect on the filter, and if the value is >150, it will have more effect on the filter, which will make the picture look like a cartoon.

Canny edge detection

Canny edge detection is a technique to extract useful structural information from different visual objects and significantly reduce the amount of data to be processed. It has been widely used in various computer vision systems.

Canny edge detection steps:

1. Gaussian filtering is applied to smooth the image in order to remove the influence of noise.

2. Calculate the gradient of the x axis and y axis of the image, and calculate the combined direction of the gradient.

3. Use non-maximum suppression to suppress those pseudo boundary points.

4. Apply min max to the image obtained above.

5. Tracking edges by hysteresis: Edge detection is accomplished by suppressing all other edges that are weak and not connected to the strong edge.

To meet the requirements Canny used a calculus of variations, which is a method of finding functions that optimize specific functions. The optimal test uses four exponential function terms, but it is very close to the first derivative of the Gaussian function.

Canny algorithm contains many parameters that can be adjusted, which will affect the calculation time and effect of the algorithm.

The size of the Gaussian filter: the first step all smoothing filters will directly affect the results of the Canny algorithm. Smaller filters also produce less blurring, which allows detection of smaller, significantly varying lines. Larger filters also blur more, painting a larger area of the image to the color value of a particular point. The upshot is that it is more useful for detecting large, smooth edges, such as the edge of a rainbow.

Dual thresholds: Using two thresholds is more flexible than using one, but it still has some common problems with thresholds. If the threshold is too high, important information may be missed. If the threshold is too low, the minor information will be regarded as important. It is difficult to come up with a universal threshold that applies to all images. There is not yet a proven implementation method.

Cornersubpix method

When we want to do geometric measurement or calibration, we need feature points with higher accuracy than target recognition. General corner detection can only obtain integer coordinate values, so we need subpixel level corner detection to obtain real coordinate values to meet the accuracy requirements.

The location of sub-pixel corner detection is a basic measurement value when the camera calibration, tracking and rebuilding the trajectory of the camera or rebuilding the 3D structure of the tracked object.

Otsu binarization method

In computer vision and image processing, the Otsu binarization method is used to automatically binalize images based on clustering, or to degrade a gray image to a binary image. The algorithm is named after Zhanyuki Otsu. The algorithm assumes that the image contains two types of pixels according to the two-mode histogram (foreground pixel and background pixel), so it needs to calculate the best threshold that can separate the two types so as to minimize their in-class variance. Since the squared distance is constant, they have the greatest variance between classes. Thus, Otsu binarization is roughly a discrete simulation of one-dimensional Fisher discriminant analysis.

The multilevel threshold extension of the original method is called the Taotsu algorithm.

In Otsu algorithm, we exhaustively search for the threshold that minimizes the in-class variance, defined as the weighted sum of the variances of two classes:

$$\sigma_w^2(t) = w_1(t)\sigma_1^2(t) + w_2(t)\sigma_2^2(t)$$

The weight is the probability of the two classes separated by the threshold t, and the variance of the two classes.

Otsu proved that minimizing intra-class variance is the same as maximizing inter-class variance:

$$\sigma_b^2(t) = \sigma^2 - \sigma_w^2(t) = w_1(t)w_2(t)[\mu_1(t) - \mu_2(t)]^2$$

In terms of class probability and class mean. Class probability is calculated by histogram with threshold t:

$$\mu_1(t) = \sum_0^t p(i)$$

And the class mean is:

$$\mu_1(t) = [\sum_{0}^{t} p(i)x(i)]$$

Where, x(i) is the value of the i-th histogram surface element center. And similarly, you can figure out the right side of the histogram for the elements greater than t. Class probability and class mean can be calculated iteratively. This idea leads to an efficient algorithm.

The Otsu algorithm yields a threshold in the 0:1 range. This threshold is used for the dynamic range of pixel intensity that appears in the image. For example, if the image contains only pixel intensities between 155 and 255, the Otsu threshold of 0.75 maps to the grayscale threshold of 230 (rather than 192 because the image does not contain the full range of pixels from 0 to 255). Common photographic images that tend to include a full range of pixel intensities make this a moot point, but other applications will be sensitive to this difference.

Finally, we use Otsu binarization method and adjust the threshold at 191, and detect the pins, moreover, we directly use this fixed parameter in opencv Otsu binarization method and get pins out, then do a combination of the original images see its difference.

Advantages: The algorithm is simple, when the area of the target and the background is not much different, it can effectively segment the image.

Disadvantages: The interclass variance method is very sensitive to noise and target size, and it can only produce good segmentation effect for the image with single peak interclass variance. When the size ratio between the target and the background is very large (for example, it is affected by the factors such as uneven illumination, reflection or complex background), the inter-class variance criterion function may present bimodal or multi-modal peaks, or the grayscale of the target and the background has a large overlap, and the effect is not very satisfactory.

Reason: This method ignores the spatial information of the image, and takes the gray distribution of the image as the basis of image segmentation, which is also very sensitive to noise.

4.5. Pins of IC chips recognition

We have two tasks to do pins of IC chips recognition. One is for IC chips with pins VS IC chips without pins.

For this task, we use SIFT to extract the feature point, then use a more statistical way, k-means to do a recognition based on many important features point. Then we get the result, use SVM to test if the IC chips with pins or without pins.

4.6. Pins of IC chips recognition: algorithms

SIFT

SIFT (Scale-invariant feature transform) is a description used in the field of image processing. This description has scale invariance and can detect key points in the image. It is a local feature descriptor.

SIFT is worthy of being called the most popular one in the aspect of feature description of digital images. Many people have made improvements to SIFT, and a series of variants of SIFT have been born. SIFT has filed for a patent.

SIFT features are points of interest based on some local appearance on the object independent of the size and rotation of the image. The tolerance of light, noise and micro Angle change is also quite high. Based on these properties, they are highly significant and relatively easy to retrieve, and in large eigenvalue databases, objects are easily identified and rarely misidentified. SIFT feature description for partial object occlusion detection rate is also quite high, even only need more than 3 SIFT object features is enough to calculate the position and orientation. With today's computer hardware speeds and small feature databases, the recognition speed can be close to real-time computing. SIFT features have a large amount of information, which is suitable for fast and accurate matching in massive databases.

SIFT algorithm has the following characteristics:

1.SIFT feature is the local feature of image, which maintains invariance for rotation, scale scaling and brightness change, and also maintains a certain degree of stability for perspective change, affine transformation and noise;

2. Tiveness is strong and informative, so it is suitable for quick and accurate matching in massive feature databases;

3. Multi-quantity, even a few objects can produce a large number of SIFT feature vectors;

4. High speed, the optimized SIFT matching algorithm can even meet the real-time requirements.

SIFT feature detection mainly includes the following four basic steps:

Scale space extremum detection: Search image locations on all scales. Gaussian differential functions are used to identify potential points of interest that are invariant to scale and rotation.

2. Locate key points

At each candidate location, the location and scale were determined by a finely fitted model. Key points are chosen according to their stability.

3. Set the direction

One or more directions are assigned to each key point location based on the local gradient direction of the image. All subsequent operations on the image data transform with respect to the direction, scale, and position of the key points, thus providing invariance to these transformations.

4. Description of key points

The local gradient of the image is measured on the selected scale in the neighborhood around each key point.

These gradients are transformed into a representation that allows for relatively large local shape deformation and illumination changes.

Cluster: K-means

K-mean clustering is an Expectation-Maximization algorithm to solve the Gaussian Mixture Model (GMM) in the normal distribution covariance is the unit matrix. The special case is obtained when the posterior distribution of the hidden variable is a set of Dirac delta functions.

1. Initialize cluster centers: C; t = 0.
2. Assign each point to the closest center:

$$\delta = Argmin \frac{1}{N} \sum_{j}^{N} \sum_{i}^{K} \delta_{ij} (C_i^{t-1} - X_j)^2$$

3. Update cluster as means of points:

$$C = Argmin \frac{1}{N} \sum_{j}^{N} \sum_{i}^{K} \delta_{ij}^{t} (C_i - X_j)^2$$

4. repeat 2-3 until no points are re-assigned. (t = t + 1)

SVM

Support Vector Machine (SVM) is a generalized linear classifier that classifies data according to supervised learning; Its decision boundary is the maximum-margin hyperplane for solving learning samples.

SVM uses hinge loss function to calculate empirical risk and adds regularization terms into the solving system to optimize structural risk. SVM is a classifier with sparse and robust. SVM is one of the common kernel learning methods, which can carry out nonlinear classification by kernel method.

k-Nearest Neighbor (KNN)

k-Nearest Neighbor (KNN) classification algorithm is a relatively mature method in theory and one of the simplest machine learning algorithms. The idea of this method is as follows: in the feature space, if most of the k nearest samples (i.e. the nearest samples in the feature space) near a sample belong to a certain category, the sample also belongs to this category.

The so-called k-nearest neighbor algorithm means that, given a training data set, for a new input instance, K instances closest to the instance are found in the training data set (that is, K neighbors mentioned above). Most of these K instances belong to a certain class, and the input instance is classified into this class.

In our 16 pins recognition, we use KNN not only for K-means, the different methods to do it.

K-means VS KNN

K-means is the most commonly used clustering algorithm, which is an unsupervised algorithm to group similar samples together. Distance is used as the evaluation index of similarity, that is, the closer the distance between two objects, the greater their similarity.

The main steps of the algorithm can be described as follows:

1. K initial clustering centers are randomly generated.
2. The distance between the test point and the cluster center is calculated, and the nearest cluster center is selected to classify the test point.
3. Update the clustering center for each class.
4. Repeat steps 2 and 3 for iterative update until the clustering center does not change or the distance between the new clustering center and the previous clustering center is less than a certain value

KNN is the simplest classification algorithm. If a sample belongs to a certain category among k samples that are most similar (i.e., closest to each other in the feature space), the sample also belongs to this category. The main steps of the algorithm can be described as follows: 1. Calculate the distance between the point in the known data set and the current point; 2. Sort by increasing distance; 3. Select k points with the minimum distance from the current point 4. Determine the occurrence frequency of k points in the category 5. Return the category with the highest frequency of k points as the prediction classification of the current point

5. Evaluation

1. For the data and image feature selection, we use a on hand way and opencv directly call the libraries and functions way. We observe that in opencv way, it is very convenient and fast. The rapid of code run faster and has more efficiency, moreover, the run out images like Gaussian filter and Canny edge detection these methods, the code is very simple, and run out images are also very nice, especially in Canny edge detection, the on hand algorithm is not as accurate as that of opencv. The opencv also show that it has a very very good robust. But for the free and flexible in use of inner algorithms, like adjust some inner parameters, the on hand way obviously better than the opencv, for example, use the Otsu method to adjust the parameter the threshold slightly. The opencv way like a black box, while on hand way like a white box.

2. For the IC chips with pins OR IC chips without pins part, we finally got the results, use data set that include train set and test set, train set has 100 nearly images, and test set has 3:7 nearly 30 images. The sample > 50, and we got the result, the accuracy is bigger than 0.5, the accuracy it is approach to 0.6, which is a good score.

3. For the IC chips that are 16 pins, not small pins, small pins, we got the accuracy 0.58, that is good, and for prediction, we finally got 0.62. For choose KNN and/OR K-means, we think it is based on the goal of the task. it is a supervised task or an unsupervised task. If it is a classification, and labeled we use a KNN, and if it is cluster, we use K-means. But all the data set need to > 50 samples.

4. For the IC chips with pins OR IC chips without pins part, we think the number of the sample is enough, not very small. So, the nearly 0.4 inaccurate we suppose is for the reason of the previouse analysis the K-means, because of the data is very big, and the machine that detect many IC chips images maybe the pins OR no pins this feature is not very accuracy, this cause the next step SVM result may loose some accuracy. We think the result is good, but need to increase the accuracy by using some additional ways.

5. For the important points detection, the SIFT has a large amount of computation and requires full parallel and full pipeline architecture for acceleration, but parallel Gaussian pyramid construction brings a lot of hardware waste. Some feature points are lost in the construction of Gaussian pyramid, which makes the feature robustness extracted by the accelerator worse.

6. For the 16 pins IC chips OR not 16 pins IC chips recognition, we think the methods we use KNN is good if the data set is large. For the whole task, we did not classify the images as that of the small sample data set we said before. On the other hand, we have tried a small sample data set that could compare with the former one. And we get a more precise and more accurate one. Because we think we have done the classification. But we think it is not a very good thing, if we still have a very high accuracy like 0.95-1.00. Because we think this small sample may not a good example in terms of not enough to be representative for the data set. That means the number of the sample maybe too small, but it has no relationship with the fit and over fit issue. So, we think some times the small sample in machine learning or deep learning is also a trend to do some research. It represents some certain phenomenon.

7. We also think the K-means may suit for the sample that is not larger than that of the KNN comparably to say. That is to say, the capacibility of classification may require more sample, and that of cluster may require a less sample data set. like for classification, the KNN can have $<$ 100K, but for cluster, the K-means can have $<$ 10K samples.

6. Conclusions

We use many different ways to do the tasks. To take the work, setting the goal the same, but use different ways to do, we think this is a very good method to see and compare directly the effects of different models and methods using in the same object.

We use the same task to do different ways in data and images feature selection, compared with the on hand and the opencv. Then, we use the almost same task to do different ways in pins recognition. compared with the K-means and KNN, also evaluate the SIFT and other methods in sample bigger than 50, and also sample smaller than 50, the small sample, which has 22 images.

For the future work, we think we need to pay more attention on the first that design the task, make a very clear and valuable points and questions, develop our critical thinking and good at asking questions and why and be critical in our observation ways.

References

Jun W, Xiaomeng Z, Jingjing W. 2021. Chip Appearance Defect Recognition Based on Convolutional Neural Network. Sensors 2021, 21(21), 7076. CHEN Y, 2020. Machine Learning and Computer Vision for PCB Verification. TRITA-EECS-EX; 2020:920.

Sean-Dar C, Ming-June T. 2011. Recognition of IC chips via machine vision. Journal of the Chinese Institute of Engineers

XingFu O, Wenfeng C, Miao Z. 2021. Design of LQFP chip pin defect detection system based on machine vision. MLISE.