Available online at www.sciencedirect.com

ScienceDirect

IFAC Papers Online CONFERENCE PAPER ARCHIVE

IFAC PapersOnLine 51-21 (2018) 76–81

Real-time Detection of Steel Strip Surface Defects Based on Improved YOLO Detection Network

First A. Jiangyun Li * **Second B. Zhenfeng Su** *
Third C. Jiahui Geng * **Corresponding author. Yixin Yin** **

* *Key Laboratory of Knowledge Automation for Industrial Processes, Ministry of Education,School of Automation & Electrical Engineering, University of Science and Technology Beijing, Beijing 100083, China (e-mail:leejy@ustb.edu.cn)*
** *School of Automation & Electrical Engineering, University of Science and Technology Beijing, Beijing 100083, China(e-mail:yyx@ies.ustb.edu.cn)*

Abstract: The surface defects of steel strip have diverse and complex features, and surface defects caused by different production lines tend to have different characteristics. Therefore, the detection algorithms for the surface defects of steel strip should have good generalization performance. Aiming at detecting surface defects of steel strip, we established a dataset of six types of surface defects on cold-rolled steel strip and augmented it in order to reduce over-fitting. We improved the You Only Look Once (YOLO) network and made it all convolutional. Our improved network, which consists of 27 convolution layers, provides an end-to-end solution for the surface defects detection of steel strip. We evaluated the six types of defects with our network and reached performance of 97.55% mAP and 95.86% recall rate. Besides, our network achieves 99% detection rate with speed of 83 FPS, which provides methodological support for real-time surface defects detection of steel strip. It can also predict the location and size information of defect regions, which is of great significance for evaluating the quality of an entire steel strip production line.

Keywords: Surface quality; Defect Detection; Steel Strip; Improved YOLO Network; Convolutional Neural Network

1. INTRODUCTION

Due to the influence of raw materials, rolling process and system control, etc., steel strip in the production process may have scars, scratches, insect prints, inclusions, bright prints, burrs, seams, black burn, iron scales, pollution and other defects. The defect images are shown in Fig. 1. These defects not only affect the steel strip surface appearance, but also damage the wear resistance, corrosion resistance, high temperature resistance and fatigue strength of the steel strip. Therefore, it is very important to detect the surface defects of the steel strip for improving the steel strip production quality.

However, there are many factors that make real-time detection of steel strip surface defects particularly difficult, such as the high-speed production line, diversity and large scale changes of defects, random distribution and non-defective interferences (oil stains and dust on the surface of steel strips).

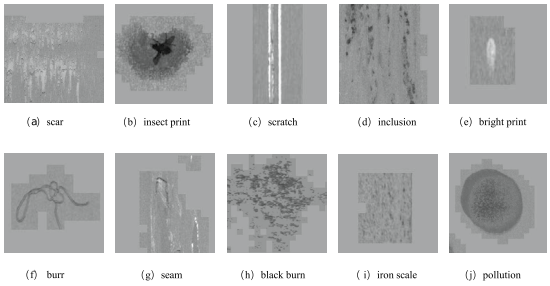


Fig. 1. Several types of surface defects on steel strip

Using Convolutional Neural Networks (CNNs), we can automatically extract multi-scale features of steel strip surface defects with good generalization and high accuracy by using a general-purpose learning procedure (LeCun et al., 2015). Using trained network, defect regions can be detected in milliseconds. Therefore, CNNs can provide an accurate, real-time detection method for surface defects in steel strip production lines, and improve the product quality of steel strips.

2. RELATED WORK

The existing surface defect detection methods are mainly based on classical machine learning algorithms, which are

⋆ This work was supported by the Fundamental Research Funds for the China Central Universities of USTB (FRF-BR-17-004A, FRF-GF-17-B49). Meanwhile, this work was also supported by the Open Project Program of the National Laboratory of Pattern Recognition (NLPR, 201800027).

10.1016/j.ifacol.2018.09.412

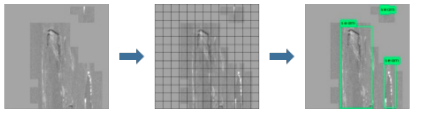


Fig. 2. YOLO flow diagram

coarsely divided into three main stages: image preprocessing, feature extraction, and classification. However, these algorithms need to design feature extractors manually, and the hand-crafted features are heavily dependent on expert knowledge and require a lot of manpower (LeCun et al., 2015)(Bengio et al., 2013). An adaptive segmentation algorithm was proposed in (Ma et al., 2017) to adaptively segment defect regions based on the gray features of the mental surface, but the types of defects cannot be distinguished. Tetrolet-based method was proposed in (Ke et al., 2016) to recognize the surface defects of steel strips. After extracting the sub-band characteristics of surface defects in different scales and directions, a Support Vector Machine (SVM) classifier was used to classify different types of surface defects. However, it took 0.239 seconds to extract features from a single defect image during testing, which is too long to meet the real-time detection requirements. Hu et al. extracted four kinds of defect features and transformed them to a 38-dimensional feature vector in (Hu et al., 2016), and an optimized SVM classifier was trained to classify 5 types of 101 defect images.

Deep multi-layer architectures of CNNs are capable of extracting more powerful features than hand-crafted features, and all of the features are extracted from training data automatically by using the backpropagation algorithm (LeCun et al., 2015)(Bengio et al., 2013). The convolutional networks provide an end-to-end solution from raw defect images to predictions, thereby alleviating the requirement to manually extract suitable features (Sermanet et al., 2013). What's more, objects can be detected in a few of milliseconds with accurate location and size information of objects via Convolutional detection networks (Redmon et al., 2016)(Redmon and Farhadi, 2016).

In view of above problems, this paper adopted You Only Look Once (YOLO) network, a convolutional detection network, to automatically extract multi-scale features of steel strip surface defects and detect the defect regions. Similar to (Springenberg et al., 2014)(Radford et al., 2015), we replaced the pooling layers with convolutional layers and made it all convolutional, allowing the network to learn its own spatial downsampling. Our network can simultaneously predict the class, location and size information of defect regions, which is very important to improve the quality of steel strips in production lines. According to our experimental results, only 12 milliseconds are needed to detect a raw defect image, which fully meets the real-time requirements of defect detection tasks in steel strip production lines.

3. NETWORK ARCHITECTURE

The YOLO detection network was first proposed by (Redmon et al., 2016) in 2015 and used for object detection tasks (Redmon and Farhadi (2016); Girshick (2015); Ren et al. (2015)). In the YOLO detection network, the convolutional layer was used to extract image features, and the softmax classifier was used to classify the object classes, the bounding boxes are predicted to locate the position of the object. In this paper, we constructed an all convolutional YOLO detection network to detect the strip surface defects, which not only improves the accuracy and detection speed, but also precisely locates the defects. What's more, we added the prediction of the surface defect size. The all convolutional YOLO detection network does not require cumbersome steps and is an end-to-end strip surface defect detection network.

Table 1. Dataset of strip surface defect images

	Scar	Scratch	Inclusion	Burr	Seam	Iron scale	Amount2
Training	673	596	572	448	591	575	3455
Test	200	200	200	200	200	200	1200
Amount1	873	796	772	648	791	775	4655

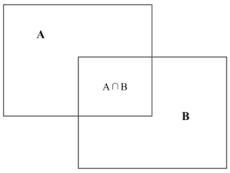


Fig. 3. IOU defines the overlap of two bounding box. IOU of rectangular box A, B calculated as $IOU = \frac{(A \bigcap B)}{(A \bigcup B)}$, which is the proportion of the overlapping area of A, B to the total area.

3.1 Detection Principal

The YOLO network divides the input image into S × S grids. Meanwhile, the convolutional layers are designed to extract the defect features. For each grid, the network determines whether the grid contains defects and identifies the defect categories according to the extracted defect features, the detection procedure is shown in Fig. 2.

The YOLO detection network will also predict B bounding boxes and the confidence of each bounding box. The bounding boxes locate the position of defects in the images, and the confidence score reflects how confident the predicted bounding box is. Formally we define the confidence score as $confidence = P_r(Object) \times IOU^{truth}_{b-box}$. $P_r(Object)$ represents the probability of defects in the grid, and IOU^{truth}_{b-box} represents the overlapping rate between the bounding box and ground truth(as shown in Fig. 3). NMS (Non-Maximum Suppression) method is adopted to remove the redundant bounding boxes.

The network predicts 5 values for each bounding box:x,y,w, h and $confidence$. The (x, y) coordinates represent the center of defect, the (w, h) represents the height and width of each box. The $confidence$ are described before.

The probability of a defect appearing in a box is defined as $P_r(Class|Object)$. At test time we multiply the class confidence score and the bounding box confidence score, defined as equation (1). The $P_r(Class_i) \times IOU^{truth}_{b-box}$ will provide class-specific confidence scores for each box. All the class confidence and the bounding boxes in each grid cell are finally encoded as $S \times S \times (5 + c) \times B$ tensor.

$$\begin{aligned} &P_r(Class_i|Object) \times P_r(Object) \times IOU_{b-box}^{truth} \\ &= P_r(Class_i) \times IOU_{b-box}^{truth} \end{aligned} \tag{1}$$

3.2 Improved YOLO Network

We constructed an all convolutional YOLO network with 27 convolutional layers. The first 25 convolutional layers are used to extract steel strip surface defect features, while the last two convolutional layers predict the defect categories and bounding boxes. The network structure is shown in Fig. 4.

Similar to (Lin et al., 2013), our network simply uses continual 3×3 convolutional layer with 1×1 reduction layer followed. The continual 3×3 filters extract the defect features from input images, and the 1×1 kernels are used to reduce the feature space of the previous feature map. Learning from (Springenberg et al., 2014)(Radford et al., 2015), max pooling layers can be replaced by convolution layers with stride of 2 without loss in accuracy on several image recognition benchmarks. Besides, the convolution layers allow the network to learn its own spatial downsampling rather than deterministic spatial downsampling. In this paper, we replaced the max-pooling functions of original network with 3×3 ($stride = 2$) convolutional functions and achieved a slightly increase in accuracy of 0.6%.

The network predicts defect categories information and bounding box information on the 13×13 feature maps ($S = 13$). This is sufficient for large-scale defects. However, some defect features will be lost during the convolution process. In addition, small-scale defects are often undetectable. To extract features from fine-grained features and improve the detection accuracy of small-scale defects, we add higher resolution feature maps to extract finer features. By using passthrough layer in (Redmon and Farhadi, 2016), the $26 \times 26 \times 512$ feature maps are transformed into the $13 \times 13 \times 2048$ feature maps and concatenated with the original $13 \times 13 \times 1024$ feature maps. The network will return the information of class and bounding box from $13 \times 13 \times 3072$ feature maps. In this paper, we mainly detect 6 types of strip surface defects and predict 5 bounding boxes in each grid, and we finally get $13 \times 13 \times 55$ tensor.

3.3 Defect database

We get the steel strip surface defect data from the cold-rolled steel strip production line. The database mainly includes scratches, inscriptions and other dozens of strip surface defect images. Due to the limited number of defect images that can be collected on the actual cold-rolled strip production line, defects of several types are too few to effectively extract defect features. Six types of defects are selected to be detected in this paper, which mainly obtain scar, scratches, inclusions, burrs, seams and iron scales defects.

Each defect image was cut into 300×300 before sent to our network, and each image has obvious defects. The surface defect database contains 6 classes of 4655 steel strip surface defects. The details of our dataset are shown in Table 1. Prior to training the network, ground truth annotations were performed on all defect images manually.

3.4 Training

The YOLO network optimized the loss function, and the effect is proved to be good. Thus we adopt the same loss function in our all convolutional network.

Loss function Aiming at the ease of optimization, the YOLO detection network uses the sum-squared error in the loss function. However, the sum-squared error weights localization error equally with classification error which does not perfectly align with the goal of maximizing average precision. In every image many grid cells dont contain any defects. This pushes the confidence scores of no-defect cells towards zero, often overpowering the gradient from those defect grid cells, which can lead the model instability.

To remedy this, the YOLO network increases the loss from bounding box coordinate predictions and decrease the loss from confidence predictions for no-defect boxes. YOLO network uses two parameters ($\lambda_{coord} = 5, \lambda_{no\_defect} = 0.5$) to accomplish this.

In order to improve the detection effect on small-scale defects, YOLO network increases the proportion of errors in the bounding box of the small-scale defects by increasing the square difference information of the width and height of the bounding box in the loss function. The optimized loss function is as follows (2):

$$\begin{aligned} Loss = \ & \lambda_{coord} \sum_{i=0}^{s^2} \sum_{j=0}^{B} 1_{ij}^{defect} [(x_i - \hat{x_i})^2 + (y_i - \hat{y_i})^2] \\ & + \lambda_{coord} \sum_{i=0}^{s^2} \sum_{j=0}^{B} 1_{ij}^{defect} [(\sqrt{w_i} - \sqrt{\hat{w_i}})^2 + (\sqrt{h_i} - \sqrt{\hat{h_i}})^2] \\ & + \sum_{i=0}^{s^2} \sum_{j=0}^{B} 1_{ij}^{defect} (C_i - \hat{C_i})^2 \\ & + \lambda_{no-defect} \sum_{i=0}^{s^2} \sum_{j=0}^{B} 1_{ij}^{no-defect} (C_i - \hat{C_i})^2 \\ & + \sum_{i=0}^{s^2} 1_{i}^{defect} \sum_{c \in classes} (p_i(c) - \hat{p_i}(c))^2 \end{aligned} \tag{2}$$

where 1_i^{defect} denotes if any kinds of defects appear in cell i and 1_{ij}^{defect} denotes that the jth bounding box predictor in cell i is responsible for that prediction.

Optimization In order to improve the accuracy and speed of defects detection, we have adopted some training strategies in the training process.

Multi-scale input. The network trained on low-resolution images has high test speed but low accuracy, while the network trained on high-resolution images shows high test accuracy but does not meet the requirement of speed. When training our network, we changed the fixed 416×416 input resolution to a variable input resolution. We set a set of selectable input resolutions {224, 256 $\cdots$ 416, 448}, and the network changes the input size every 10 iterations. Such strategy can ensure that the network extracts features from images of different scales, and the

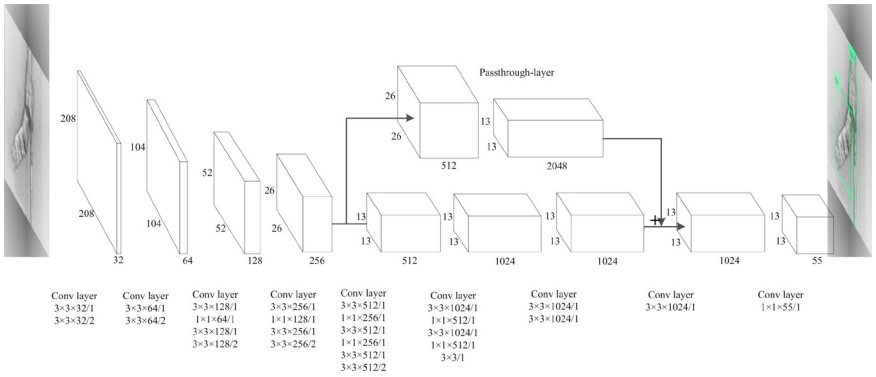


Fig. 4. Architecture of improved YOLO detection network.

detection results can be traded between accuracy and speed.

Batch Normalization. In the process of training the network, the network does not train all the images at the same time, but first divides all the images into several batches. Our network utilizes Batch Normalization (Ioffe and Szegedy, 2015) to normalize the data for each batch, as shown in (3). Here x_i denotes the activation input, and batch size is m.

$$\begin{aligned} \mu_\beta &\longleftarrow \frac{1}{m}\sum_{i=1}^{m} x_i \\ \sigma_\beta^2 &\longleftarrow \frac{1}{m}\sum_{i=1}^{m}(x_i-\mu_\beta)^2 \\ \hat{x}_i &\longleftarrow \frac{x_i-\mu_\beta}{\sqrt{\sigma_\beta^2+\varepsilon}} \\ y_i &\longleftarrow \gamma\hat{x}_i+\beta \equiv BN_{\gamma,\beta}(x_i) \end{aligned} \tag{3}$$

Batch normalization leads to a significant improvement in convergence while eliminating the need for other forms of regularization. It also helps regularize the model. By adding batch normalization on all of the convolutional layers we get more than 2% improvement in mAP.

Multi-scale features. The YOLO extracts defect features directly from the entire image, making full use of the contextual information of the original image to ensure a high recall rate. In our network, feature maps of different scales are jointly connected to predict the classification information and bounding boxes, which reduce the loss of information in defect images (Lin et al., 2017).

Data augmentation. When the training data is not sufficient, data augmentation (Krizhevsky et al., 2012) can expand the data set and increase the diversity of the training data. Data augmentation can also reduce overfitting. Before we train the network, we performed sharpness augmentation and contrast augmentation on some of the defect images. During the training process, we randomly scaled and cropped the defect images.

Activation function. We use a linear activation function for the final layer and all other layers use the following leaky rectified linear activation (4):

$$\varphi(x) = \begin{cases} x & x > 0 \\ 0.1x & \text{otherwise} \end{cases} \tag{4}$$

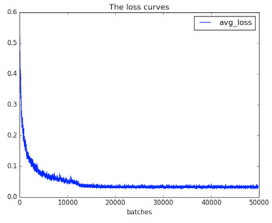


Fig. 5. The loss decay curve.

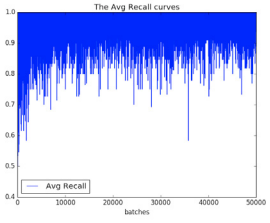


Fig. 6. The recall decay curve.

4. EXPERIMENTS

We trained the improved YOLO network for 50,000 iterations on the 6 types of defect images. Throughout training we use stochastic gradient descent with a batch size of 64, a momentum of 0.9 and a decay of 0.0005. The learning rate was initialized at 0.01 and was divided by 10 after every 12000 iterations. The training process took 12 hours on two NVIDIA GTX 1080Ti GPUs. Dealing with the training process data, we get the loss attenuation curve and recall curve, as shown in Fig. 5 and Fig. 6.

The test defect images were detected with a trained network and 1200 images were completed within 13 seconds, achieving 97.55% mAP and 95.86% recall rate. The detection details are shown in Table 2, and the detection results are shown in Fig. 7.

Table 2. Detection results (mAP & Recall)

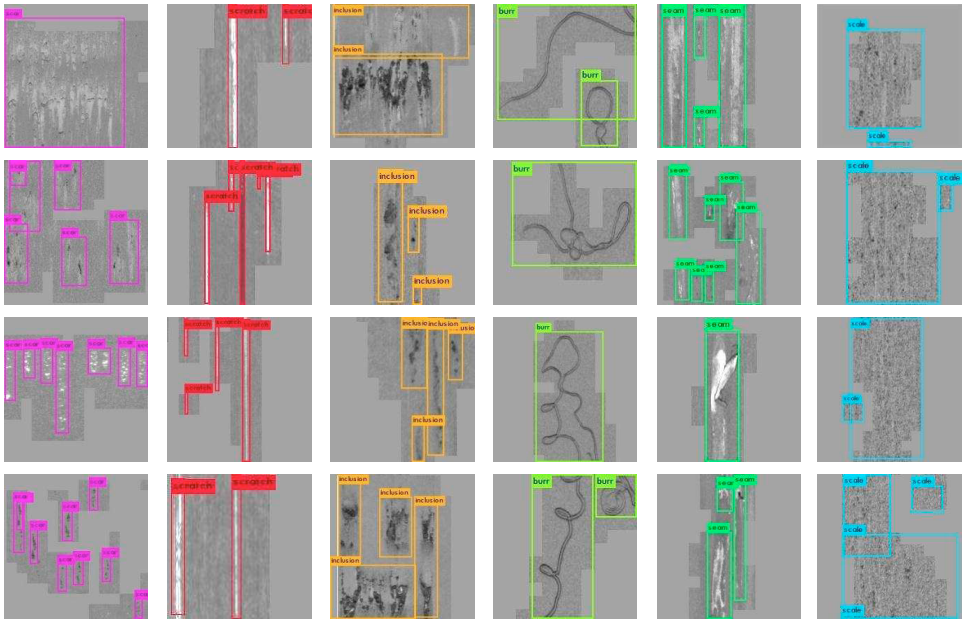
	Scar	Scratch	Inclusion	Burr	Seam	Iron scale	Avg.
mAP	97.54%	99.02%	97.20%	97.56%	97.07%	97.45%	97.55%
Recall	92.29%	98.70%	95.29%	99.17%	93.17%	99.26%	95.86%

4.1 Comparison with traditional methods

Traditional algorithm mainly focused on the defect classification problem, and they usually cannot solve the problem of locating defects and predicting the size of defects, such as the SVM classification in (Hu et al., 2016) and some machine learning algorithms in (Ke et al., 2016)(Guo et al., 2017). Our improved YOLO detection network can not only classify defect images, but also accurately obtain the position and size information of defects. We compared our

Table 3. Comparisons with other methods

	Task	Accuracy	Inference time per image	Amount of dataset
M-pooling CNN(Masci et al., 2012)	classification	93.03%	unknown	2927
HCGA(Hu et al., 2016)	classification	95.04%	0.158s	351
HSVM-MC(Chu et al., 2017)	classification	95.18%	1.1044s	900
Infrared imaging(Zhang, 2011)	classification	95.42%	unknown	1200
Contourlet transform(Xu et al., 2013)	classification	96.46%	0.103s	868
Tetrolet transform(Ke et al., 2016)	classification	97.38%	0.239s	868
Ours	classification + location + scale	97.55%	0.012s	4655



(a) scar (b) scratch (c) inclusion (d) burr (e) seam (f) iron scale

Fig. 7. Detection results of 6 types of surface defects on steel strip.

detection results with traditional methods, the results are shown in Table 3.

As can be seen in Table 3, the YOLO detection network is significantly higher in classification accuracy than the shallow neural network in (Masci et al., 2012) and the SVM classifier in (Chu et al., 2017), slightly higher than the Tetrolet transform in (Ke et al., 2016) and the hybrid chromosomal genetic algorithm in (Hu et al., 2016). Since the number of defect images in this experiment far exceeds the number in these referrences, the defect features that our network extracted have better generalization.

4.2 Real-time analysis

The average inference time for our network to detect a strip surface image is only 0.012s. It means our network can detect 83 defect images per second, which is more than ten times or even dozens times faster than methods in (Ke et al., 2016)(Hu et al., 2016). The maximum speed of the actual production line is 30m/s and the view field of a single camera is 50-100cm. This requires the defect detector must have a speed of 30-60 FPS. Our improved YOLO detection network in this paper achieves a detection speed of 83 FPS, which fully meets the real-time detection speed requirements of the actual production lines.

4.3 The detection rate

When only needs to detect the defects without classifying the categories, our network achieves 99% detection rate, with only 1% defects missed. With the growth and accumulation of online defect data, the performance of our network can be further improved.

4.4 Defect scale accuracy

According to the experiments, our network can detect defects with a minimum area of 10 square millimeters.

4.5 Future outlook

Deep learning is a data-driven learning method, and the amount of data sets directly affects the learning results. If a larger number and variety of strip surface defects images are available to train our network, the network will have better performance and higher accuracy. Meanwhile, the location and scale of the defects will be more accurate.

5. CONCLUSIONS

We established a steel strip surface defect database which contains surface defects of six types of cold-rolled steel strip. We detected the surface defects by constructing an

all convolutional YOLO detection network. The results show that our network achieves a 97.55% mAP, 95.86% recall rate and 99% detection rate. The network provides an end-to-end detection solution for strip surface defect, and achieves a detection speed of 83 FPS, making the real-time detection of strip surface defects more effective.

The improved YOLO detection network can predict location and the scale information of surface defects on the entire strip production line, which is of great significance for improving the product quality of the strip steel production. In the case of obtaining more types and quantities of strip surface defect data, this method can be further improved in detection accuracy.

REFERENCES

Bengio, Y., Courville, A., and Vincent, P. (2013). Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35(8), 1798–1828.

Chu, M., Zhao, J., Gong, R., and Liu, L. (2017). Steel surface defects recognition based on multi-label classifier with hyper-sphere support vector machine. In *Control And Decision Conference (CCDC), 2017 29th Chinese*, 3276–3281. IEEE.

Girshick, R. (2015). Fast r-cnn. *arXiv preprint arXiv:1504.08083*.

Guo, H., Shao, W., and Zhou, A. (2017). Novel defect recognition method based on adaptive global threshold for highlight mental surface. *Chinese Journal of Scientific Instrument*, 38(11), 2797–2804.

Hu, H., Liu, Y., Liu, M., and Nie, L. (2016). Surface defect classification in large-scale strip steel image collection via hybrid chromosome genetic algorithm. *Neurocomputing*, 181, 86–95.

Ioffe, S. and Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, 448–456.

Ke, X.U., Lei, W., and Wang, J. (2016). Surface defect recognition of hot-rolled steel plates based on tetrolet transform. *Journal of Mechanical Engineering*.

Krizhevsky, A., Sutskever, I., and Hinton, G.E. (2012). Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, 1097–1105.

LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *nature*, 521(7553), 436.

Lin, M., Chen, Q., and Yan, S. (2013). Network in network. *arXiv preprint arXiv:1312.4400*.

Lin, T.Y., Dollár, P., Girshick, R., He, K., Hariharan, B., and Belongie, S. (2017). Feature pyramid networks for object detection. In *CVPR*, volume 1, 4.

Ma, Y., Li, Q., He, F., Yan, L., and Xi, S. (2017). Adaptive segmentation algorithm for metal surface defects. *Chinese Journal of Scientific Instrument*.

Masci, J., Meier, U., Ciresan, D., Schmidhuber, J., and Fricout, G. (2012). Steel defect classification with max-pooling convolutional neural networks. In *Neural Networks (IJCNN), The 2012 International Joint Conference on*, 1–6. IEEE.

Radford, A., Metz, L., and Chintala, S. (2015). Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*.

Redmon, J., Divvala, S., Girshick, R., and Farhadi, A. (2016). You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 779–788.

Redmon, J. and Farhadi, A. (2016). Yolo9000: better, faster, stronger. *arXiv preprint*, 1612.

Ren, S., He, K., Girshick, R., and Sun, J. (2015). Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, 91–99.

Sermanet, P., Eigen, D., Zhang, X., Mathieu, M., Fergus, R., and LeCun, Y. (2013). Overfeat: Integrated recognition, localization and detection using convolutional networks. *arXiv preprint arXiv:1312.6229*.

Springenberg, J.T., Dosovitskiy, A., Brox, T., and Riedmiller, M. (2014). Striving for simplicity: The all convolutional net. *arXiv preprint arXiv:1412.6806*.

Xu, K., Ai, Y.H., Zhou, P., and Yang, C.L. (2013). Recognition of surface defects in continuous casting slabs based on contourlet transform. *Journal of University of Science Technology Beijing*, 35(9), 1195–1200.

Zhang, X. (2011). Vision inspection of metal surface defects based on infrared imaging. *Acta Optica Sinica*, 31(3), 0312004.