Multimed Tools Appl
https://doi.org/10.1007/s11042-017-5268-7

Real-time eye detection in multimedia environment

Zepeng Wang[1] · **Luming Zhang**[1]

Received: 15 September 2017 / Revised: 27 September 2017 / Accepted: 29 September 2017

Abstract Eye detection is more and more important in many applications of Multimedia Environment, such as face detection, video surveillance, emotional analysis. Many methods relay on some expensive hardware, and they are also complicated. So in this paper, we propose a simple method only with a webcam to realize real-time eye detection. First, we seize human face from the webcam. And our program will search eyes automatically. Then, we convert eyes' image to grayscale image in order to post-processing. We use Gaussian smoothing to reduce noise and Histogram equalization to enhance contrast. Finally, the image is binarized so the eye features are more prominent. And we can use Hough transform to draw a circle that fits the eyeball, so the center of the circle is the pupil. The experiment verifies the effectiveness of our method.

Keywords Eye detection · Gaussian smoothing · Histogram equalization · Hough transform

1 Introduction

Over the past few years, eye detection is a meaningful work in many applications. As a very prominent feature of the face, eye detection is very important for face detection, gaze estimation, emotion analysis. Eye detection has attracted the research of face detection [1]. We can estimate a person's gaze through eye detection [19]. Person's emotion can also be estimated by observing his eyes.

Some researchers use hardware to detect eye, like IR-illumination rigs or Kinect sensor [7]. And some applications may require people to wear cumbersome head-mounted equipment to communicate with computer. Obviously, it is not convenient. So we propose a simple method only with a webcam to detect eye. We use Hough transform to fit the eyeball with a circle, and the center of the circle is the pupil.

✉ Zepeng Wang
1422617933@qq.com

[1] Department of CSIE, Hefei University of Technology, Hefei, China

2 Previous work

Eye detection has become more and more popular in the last few years. Many researchers presented a variety of methods to realize eye detection.

Jie Zhu and Jie Yang presented an analytical method that uses edges and local patterns to obtain detection of eye features with subpixel precision [19]. They proposed new method for detecting the inner eye corner and the center of the iris, and they applied these new methods in detecting gaze tracking.

Mingli Song et al. presented a method named visual-context pattern(VCP) that used features surrounding eyes [6]. This method can make the best of eye features with some stable region on the face instead of only the region of eye.

Zhiheng Niu et al. presented a novel classifier designing framework named 2D cascaded AdaBoost to apply to eye localization [5]. The 2D structure has many advantages, for example, it greatly facilita tes the classifier designing on huge-scale training set. It can also deal with the significant variations within the positive or negative samples.

Peng Wang and Qiang Ji presented a statistical learning method to extract features and construct classifiers for multi-view face detection and eye detection [9]. The method performs well both in face detection and eye detection.

Oliver Jesorsky et al. presented a method to detect face using the Hausdorff distance [3]. This method used as a similarity measure between a general face image mode and a test image. So in the same way, we can also use this method to detect eyes.

Mark Everingham and Andrew Zisserman [2] addressed some tasks of localizing the eyes in face images. They compared three methods and found that the simple Bayesian approach performed best on the same databases.

Combined with the experience of the above researchers, we put forward our own method.

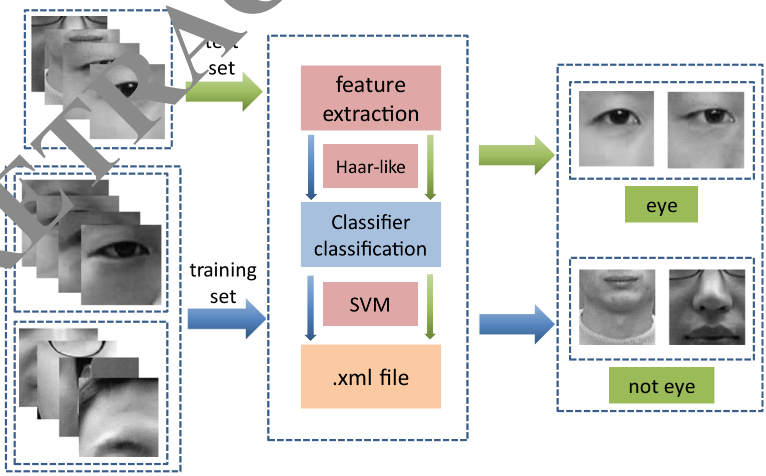


Fig. 1 The flowchart of our method to train our database. We use Haar-like to extract eye features. And our program will find eyes automatically from the webcam. When find eyes, it will draw a bounding box for each eye

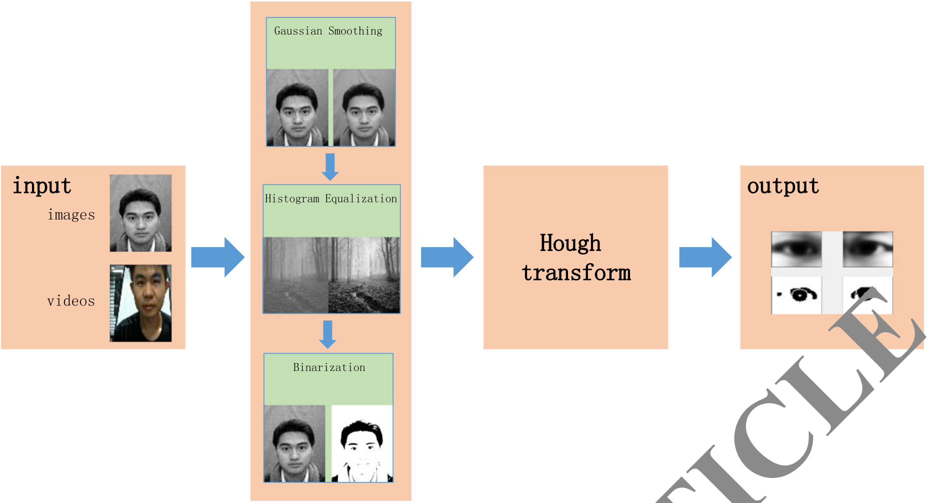


Fig. 2 The flowchart of our method

3 The proposed approach

In this paper, we propose a simple method that detects eyes easily. The flowchart of our method to train our database is shown in Fig. 1.

The flowchart of our method is shown in Fig. 2.

3.1 Haar-like features

We use Haar-like features in eye detection. Haar-like features reflect the gray scale of the image. For example, the region of eye is darker than its surrounding region. Paul Viola and Michael J. Jones proposed a new method called "integral image" that can process images extremely rapidly [3]. So we employ "integral image" to compute Haar-like features. Four example rectangle features are shown in Fig. 3.

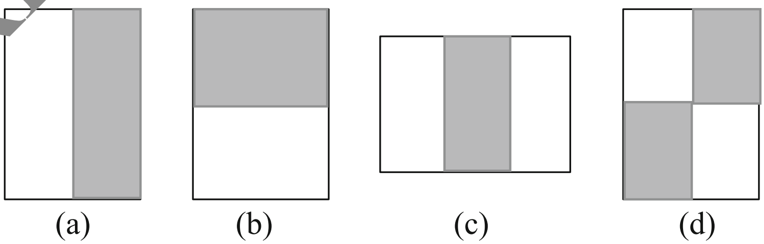


Fig. 3 Four example rectangle features. The figure that the sum of pixels in the grey rectangles subtract the sum of pixels in the white rectangles is the value of features

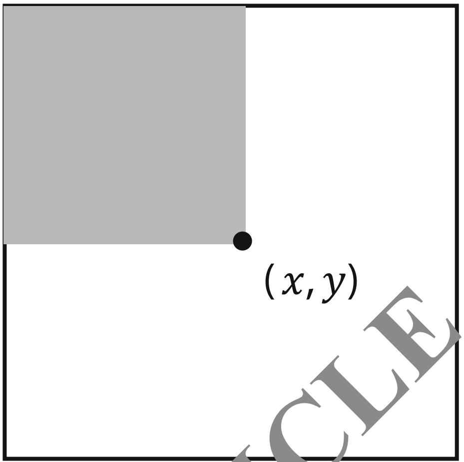


Fig. 4 The value of integral image of (*x, y*) is the above and to the left of (*x, y*)

Images can be processed very rapidly using integral image. As shown in Fig. 4, the integral image at location *x, y* contains the sum of the pixels above and to the left of *x, y* [8]. The formula is:

$$ii(x,y) = \sum_{x' \le x, y' \le y} i\left(x', y'\right)$$

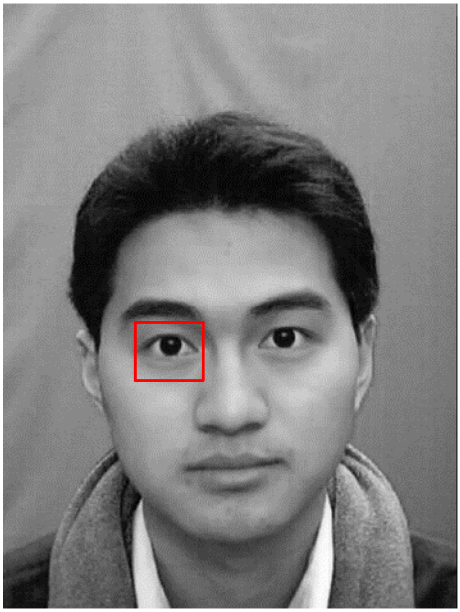


Fig. 5 We choose left eye of every image as positive samples

Fig. 6 A typical template of Gaussian smoothing

$\frac{1}{273}$

1	4	7	4	1
4	16	26	16	4
7	26	41	26	7
4	16	26	16	4
1	4	7	4	[illegible]

We choose 800 positive images with different genders, ages, lighting condition. Positive images are chosen from CAS-PEAL which is a database of face images made by Institute of Computing Technology, Chinese Academy of Sciences. As shown in Fig. 5, we choose left eye of every image as positive samples.

3.2 Gaussian smoothing

When we get the bounding boxes of eyes, we convert color images to greyscale images. Then we employ Gaussian smoothing to reduce noise.

The principle of Gaussian smoothing is that we use a template to scan every pixel in an image, and we calculate the weighted average of pixels in the neighborhood of the template. Then we use the value that we obtain to replace the value of pixel that locate in the center of the template. A typical template is shown in Fig. 6.

As shown in Fig. 7, Gaussian smoothing can reduce noise.

(a) (b)

Fig. 7 Image (**a**) is the original image, and image (**b**) is the Gaussian smoothed image

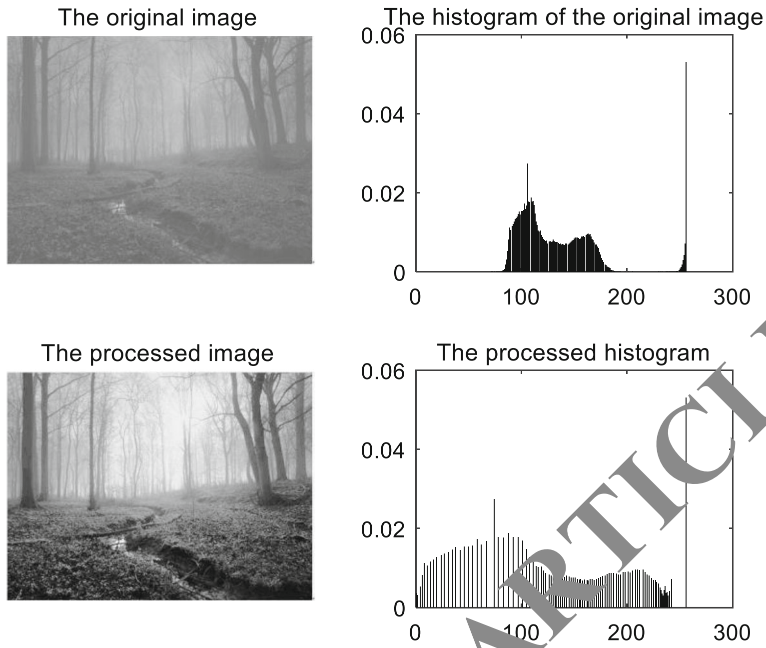


Fig. 8 The original image and the processed image. We can see the histogram of the processed image has a uniform distribution

3.3 Histogram equalization

We employ Histogram equalization to enhance contrast. The principle of Histogram equalization is that the gray histogram of the original image is changed from a clustered gray scale to a uniform distribution over the entire gray scale [4].

As shown in Fig. 8, the original image is processed used Histogram equalization. We can see the processed image has better contrast.

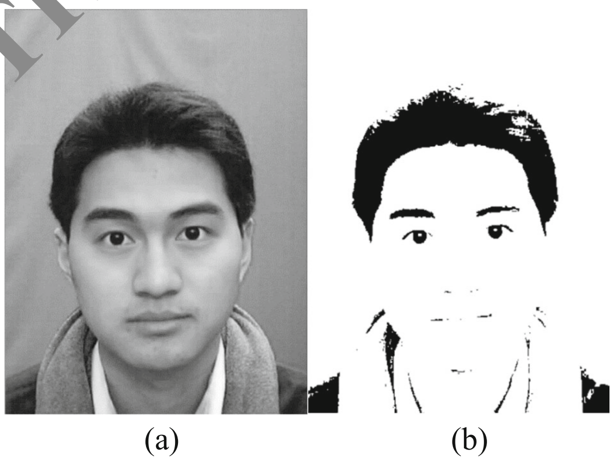


Fig. 9 Image (**a**) is the original image, image (**b**) is binarization

(a) (b)

Fig. 10 Image (**a**) is the original image, image (**b**) is processed by Canny Edge Detector after binarization

3.4 Binarization

In order to highlight features of eye, we employ binarization. We set a threshold. When the value of pixel greater than threshold, we set it to 255. And when the value of pixel less than threshold, we set it to 0. In this way, the image becomes a black and white image, so the eyeball is more prominent.

As shown in Fig. 9, we can see eyeball is more prominent after binarization.

3.5 Hough transform

The Hough transform is a method for finding lines, circles or other simple forms in an image [4]. After binarization, eyeball is more prominent. So we employ Hough transform to draw a circle that fits the eyeball. Before that, we can employ the Canny Edge Detector to extract edges. In this way, the circle that drawn by Hough transform will be more accurate.

As shown in Fig. 10, we extract edges using Canny Edge Detector.

4 Experimental results and analysis

In this section, we validate feasibility, effectiveness, and stability of our method through experiment. We separate the left eye and right eye for more precise handling. And we set two buttons for left eye and right eye to adjust threshold in order to choose suitable parameters. The boundary of the circle is the eyeball, so the center of the circle is the pupil.

Table 1 The result of our image experiment

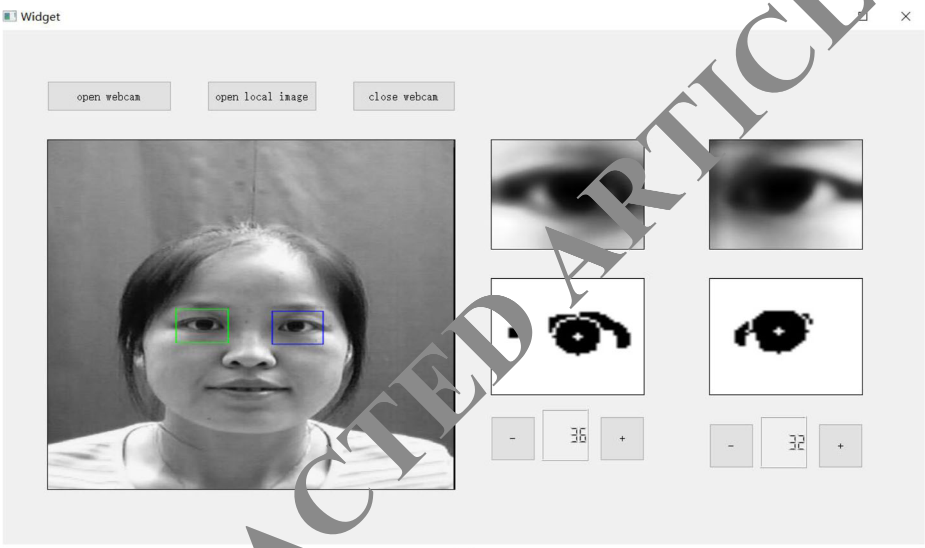
Grey images	Accurately identify images	Correct rate
50	48	96.0%
80	75	93.7%
100	88	88.0%

Table 2 The result of our video experiment

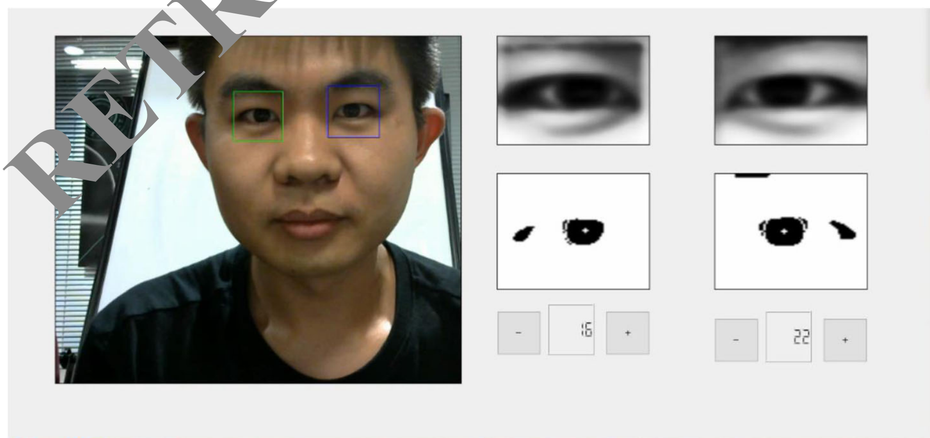
Videos	Accurately identify of the time	Correct rate
10s	9 s	90.0%
20s	17 s	85.0%
30s	25 s	83.3%

We choose 50, 80, 100 grayscale images for testing, test images are also chosen from CAS-PEAL, and the result is shown in Table 1.

We recorded 10 s, 20 s, 30 s of videos. The result is shown in Table 2.



(a)



(b)

Fig. 11 Our experiment about images and videos

Fig. 12 Different face poses in our system. Image (**a**) is rotation in the *x-y* plane. Image (**b**) is rotation in the *x-z* plane

The image experiment is shown in Fig. 11a, and the video experiment is shown in Fig. 11b. Our interface is made with QT software.

As shown in Fig. 12, we discover that our system has a certain robustness to rotation.

Many researchers realized eye detection, so we do some researches to compare different methods. In the paper [6], authors employed a normalized error for comparison between different methods. As shown in Fig. 13, compared with Niu [5] and Wang [9], our method is slightly less. But as we enhance our positive images appropriately, our system will be better.

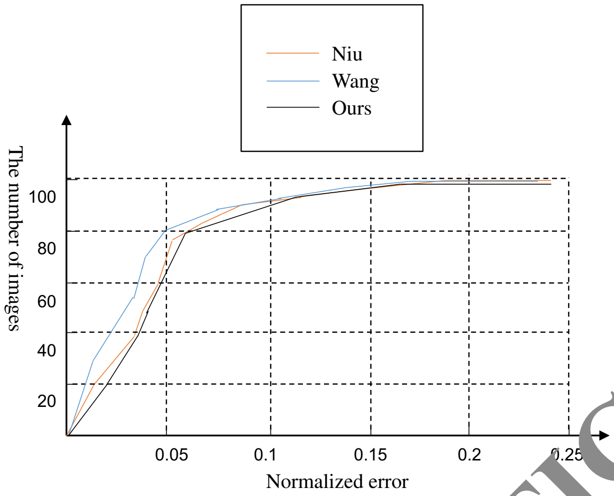


Fig. 13 Comparison of eye-detection methods on our collected image set

The advantage of our method is easy to implement. Only with a webcam, we can detect eyes. Our system runs on a platform of Inter i7 at 2.8GHz with 8-GB memory. As long as we adjust to the appropriate parameters, our system is effective. Our experiment of eye detection on the webcam-based video shows that our system can detect eyes in real time.

The experiment validates our method is feasible, effective. And when we choose suitable parameters, our system is also stable. The experiment shows that when we test images, the correct rate reaches 92.6%, and the correct rate reaches 86.1% when we test videos captured from webcam.

5 Conclusion

In this paper, we propose our simple method for eye detection. Eye detection is an important application in computer vision and graphics [10–18]. We employ Gaussian smoothing to reduce noise, histogram equalization to enhance contrast, and Hough transform to draw a circle that fits eyeball. Our method is simpler and more convenient compared with the methods that need hardware to detect eyes. Experiments show that our method is feasible, effective and stable. And the correct rate is more than 82% for detecting eyes.

References

1. Baltrusaitis T, Robinson P, Morency L-P (2016) OpenFace: An open source facial behavior analysis toolkit. Applications of Computer Vision (WACV), 2016 IEEE Winter Conference on, IEEE, Lake Placid, 7-10 March 2016
2. Everingham M, Zisserman A (2006) Regression and classification approaches to eye localization in face images. Automatic Face and Gesture Recognition, 2006. FGR 2006. 7th International Conference on, IEEE, Southampton, 10-12 April 2006
3. Jesorsky O, Kirchberg KJ, Frischholz RW (2001) Robust face detection using the Hausdorff distance. Springer, lecture notes in computer science, LNCS-2091, pp. 90-95, Halmstad, Sweden, 6–8
4. Kaehler A, Bradski G. Learning OpenCV3. pp. 349–358

5. Niu Z, Shan S, Yan S, Chen X, Gao W (2006) 2D cascaded adaboost for eye localization. Pattern Recognition, 2006. ICPR 2006. 18th International Conference on, IEEE, Hong Kong, 20-24 Aug. 2006
6. Song M, Tao D, Sun Z, Li X (2010) Visual-context boosting for eye detection. IEEE Trans Syst Man Cybern B Cybern 40(6):1460–1467
7. Sun L, Song M, Liu Z, Sun M-T (2014) Real-time gaze estimation with online calibration. IEEE MultiMedia 21(4):28–37
8. Viola P, Jones MJ (2004) Robust real-time face detection. J Int J Comput Vis 57(2):137–154
9. Wang P, Ji Q (2007) Multi-view face and eye detection using discriminant features. Comput Vis Image Underst 105(2):99–111
10. Zhang Luming, Han Y, Yang Y, Song M, Yan S, Tian Q (2013) Discovering discriminative Graphlets for aerial image categories recognition. IEEE Trans Image Process (T-IP) 22(12):5071–5084 (IF:3.199, CCF A, JCR 2)
11. Zhang Luming, Song M, Qi Z, Liu X, Jiajun B, Chen C (2013) Probabilistic Graphlet transfer for photo cropping. IEEE Trans Image Process (T-IP) 21(5):803–815 (IF:3.199, CCF A, JCR 2)
12. Zhang Luming, Gao Y, Ji R, Dai Q, Li X (2014) Actively learning human gaze shifting paths for photo cropping. IEEE Trans Image Process (T-IP) 23(5):2235–2245 (IF:3.199, CCF A, JCR 2)
13. Zhang Luming, Gao Y, Zimmermann R, Qi T, Li X (2014) Fusion of multi-channel local and global structural cues for photo aesthetics evaluation. IEEE Trans Image Process (T-IP) 23(3):1419–1429 (IF:3.199, CCF A, JCR 2)
14. Zhang Luming, Yi Y, Gao Y, Wang C, Yi Y, Li X (2014) A probabilistic association model for segmenting weakly-supervised images. IEEE Trans Image Process (T-IP) 23(9):4150–4159 (IF:3.199, CCF A, JCR 2)
15. Zhang Luming, Gao Y, Ji R, Lv K, Shen J (2014) Representative discovery of structure cues for weakly-supervised image segmentation. IEEE Trans Multimedia (T-MM) 16(2):470–479 (IF:1.754, CCF B, JCR 2)
16. Zhang Luming, Song M, Yi Y, Qi Z, Chen Z, Sebe N (2014) Weakly supervised photo cropping. IEEE Trans Multimedia (T-MM) 16(1):94–107 (IF:1.754, CCF B, JCR 2)
17. Zhang Luming, Xia Y, Ji R, Li X (2015) Spatial-aware object-level saliency prediction by learning Graphlet hierarchies. IEEE Trans Ind Electron (T-IE) 62(2):1301–1308 (IF: 5.165, JCR 1)
18. Zhang Luming, Gao Y, Xia Y, Dai Q, Li X (2015) A fine-grained image categorization system by Cellet-encoded spatial pyramid modeling. IEEE Trans Ind Electron (T-IE) 62(1):564–571 (IF: 5.165, JCR 1)
19. Zhu J, Yang J (2002) Subpixel eye gaze tracking. Automatic Face and Gesture Recognition, 2002. Proceedings. Fifth IEEE International Conference on, IEEE, Washington, DC, 21-21 May 2002

Zepeng Wang received his Bachelor degree in Hefei University of Technology,China in 2017. And he is currently studying for a master's degree at Hefei University of Technology.