Robust Piano Music Transcription Based on Computer Vision

Jun Li
Huazhong University of Science and Technology, Wuhan, China
yxlijung@gmail.com

Wei Xu
Huazhong University of Science and Technology, Wuhan, China
xuwei@hust.edu.cn

Yong Cao
Huazhong University of Science and Technology, Wuhan, China
ccvey789@gmail.com

Wei Liu
Huazhong University of Science and Technology, Wuhan, China
liuwei@hust.edu.cn

Wenqing Cheng
Huazhong University of Science and Technology, Wuhan, China
chengwq@mail.hust.edu.cn

ABSTRACT

Recently, automatic music transcription aiming to convert acoustic music signals into symbolic notations attracts increasing attention. In order to deal with the challenges of automatic music transcription based on acoustic information, traditional approaches adopt hough transform to locate the piano keyboard and a weak classifier to detect pressed keys. However, the hough transform and weak classifier show insufficient detection ability in the changing environment. In this paper, we devise a robust visual piano transcription system using semantic segmentation for the piano keyboard detection and a CNN-based classifier to detect the pressed keys, which improves the frame-level transcription results. In addition, in view of lacking public datasets in the field of visual piano transcription, we further propose a new dataset for visual piano transcription. To demonstrate the effectiveness of our system, we evaluate it on both the published dataset and we proposed, and our system significantly outperforms the state-of-the-art approaches.

CCS Concepts

• **Computing methodologies → Artificial intelligence → Computer vision→Computer vision tasks.**

Keywords

Automatic Music Transcription; Convolutional Neural Network; Semantic Segmentation; Computer Vision

1. INTRODUCTION

Automatic music transcription is the process of converting acoustic music signals into some form of symbolic notations, such as music score, musical instrument digital interface files(MIDI) [1][2]. The core task of AMT is an estimate of concurrent pitches overlap each other at the same time, so it is difficult to obtain accurate recognition results only by analyzing the audio. To solve this problem, many studies have adopted computer vision-based approaches.

Previous vision-based approaches are poorly robust and can't automate transcription. In[3], we need to manually select the background image that contains only the piano, the algorithm has low accuracy and the F1 Score is 0.68. It performs worse for more complex and faster tunes. The Clavison[4][5] transcription system proposed by Akbari et al. decreases the white key's F1 Score by 0.29 and the black key's F1 Score by 0.44 under a non-ideal environment. Akbari et al's transcription system[6] uses a piano keyboard detection algorithm based on hough transform, which can't accurately detect piano keyboard under uneven lighting or lens distortion, and the proposed CNN-SVM classifier is not trained end-to-end, meanwhile the CNN network structure is relatively simple.

This paper presents a more robust piano polyphonic transcription system based on computer vision. We use semantic segmentation to detect piano keyboard, which can accurately detect keyboard under uneven lighting and lens distortion. At the same time, we propose an automatic background selection algorithm based on hand and keyboard semantic segmentation, which enables the system to automate transcription. In addition, we design a CNN model that is more suitable for detecting pressed keys and outperforms the state-of-the-art approaches. The only publicly available dataset in the field of visual piano transcription is [6]. However, the video resolution is low and test set is simple on this dataset, which is difficult to evaluate the performance of the algorithm in actual performance. Therefore, we create a new dataset that contains videos of different scenes and playing difficulties, which is of great significance for evaluating the transcription system. Our system outperforms the state-of-the-art approaches on each dataset. Our algorithm achieves a higher accuracy with an average F1 Score of 0.965 on the dataset[6] and a high accuracy with an average F1 Score of 0.93 on our proposed dataset.

The rest of this paper is organized as follows: The second chapter introduces related research on vision-based automatic music transcription. The third chapter describes our system architecture and detailed implement. The fourth chapter describes the experimental setup and experimental results of system, which proves the advancedness of our proposed approach through comparative experiments. The fifth chapter summarizes the work of this paper and points possible directions for further studies.

2. RELATED WORK

The related researches about AMT can date back to 1977[7], A large number of scholars have made great progress in AMT for four decades. Visual music transcription mainly includes two aspects, one is a purely visual transcription method, and the other is

HPCCT & BDAI 2020, July 3–6, 2020, Qingdao, China
© 2020 Association for Computing Machinery.
ACM ISBN 978-1-4503-7560-3/20/07…$15.00
https://doi.org/10.1145/3409501.3409540

a multi-modal transcription method in which audio and video are fused.

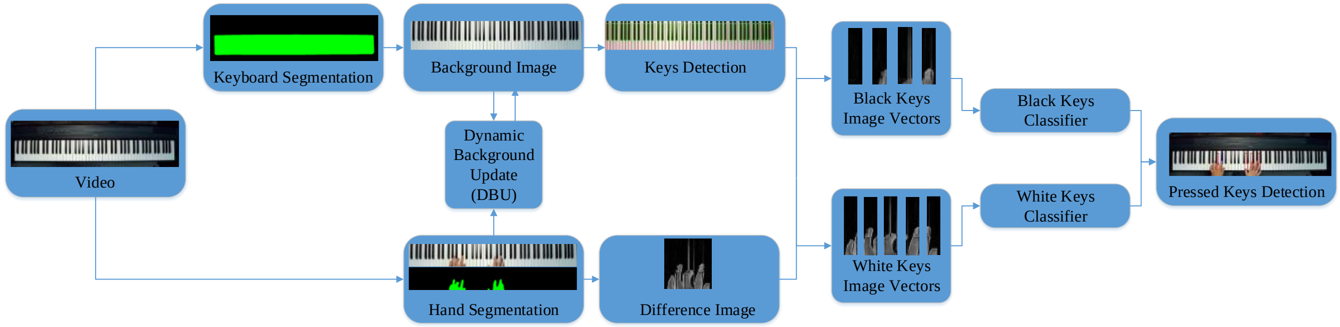

Figure 1 The overall framework of the more robust AMT approach proposed in this paper

2.1 Visual Music Transcription

In order to be able to analyze piano music transcription based on visual information, the position of piano keyboard and each key should be detected. From the fact that piano keyboard is rectangular and that each piano key is either white or black, Goodwin and Green[8] demonstrate that piano keyboard and each key can be detected using computer vision algorithms such as hough transform and binarization. Vishal[9] proposes that fingers can be tracked using shadow made by artifical light directing on the keyboard. Suteparuk [3] proposes that the difference between the video frame and the background image can be used for piano transcription, but the method requires manual selection of the background image. Akbari et al. propose the claVision[4][5] transcription system, which transcribes music based on changes in key edge strength, and F1 Score is 0.95 under certain conditions. But this method has many limitations, such as the pressed keys can't be detected accurately when the light is too dark, too bright or a large part of the key is covered by hands. Subsequently, Akbari et al. propose a visual transcription system using the CNN-SVM classifier[6] which can work in suboptimal environment, but the piano keyboard detection algorithm based on hough transform is not robust, and the CNN structure is simple. Recently, Kang[10] combines 3D CNN to solve the problem of the intensity of pressed keys by combining spatial and temporal information.

2.2 Audio-Visual Music Transcription

Recently, researchers have begun to use audio-visual information for multi-modal transcription system. Methods based on audio transcription are affected by noise and frequency multiplication, methods based on vision are difficult to solve the problem of hand-covered keys, multi-modal fusion can make up for the shortcomings of audio and video. Frisson[11] implements a multi-modal system for extracting information from guitar music. Paleari[12] implements a complete multi-modal fusion approach for guitar music transcription in 2008, which has a 89% accuracy in detecting the notes. Wan et al.[13][14] propose an automatic piano transcription approach that uses both audio and video fusion. Firstly detecting the position of the pianist's hand on the keyboard, then using the audio to transcribe pitches, and finally removing pitches that are't in the range of the hand. Lee et al.[15] propose a multi-task learning audio-video fusion framework using two-stream CNN, which achieves a transcription accuracy of 85.69%.

3. APPROACH

Figure 1 is the overall framework of the more robust AMT approach proposed in this paper, which includes four main stages to process music transcription: piano keyboard registration, hand detection, dynamic background update, and pitch detection. In this chapter, we will introduce the details of each step.

3.1 Keyboard Registration

The piano keyboard registration includes keyboard detection, automatic background image selection, key detection.

We need to detect the piano keyboard accurately. Object detection is not suitable for keyboard detection, and the piano keyboard is semantically consistent, so it is very suitable to use semantic segmentation for pixel-by-pixel prediction to accurately locate the four corner points of the keyboard. We use the PSPNet[16] semantic segmentation model which uses global context information and context aggregation in different region to segment image. Keyboard segmentation is pixel-level two classification task. The task is simple and can achieve high accuracy, which can meet the detection requirements in various environments.

Algorithm 1 Automatic Background Selection Algorithm

```
Input: F = {$f_1$,...,$f_n$}
    F is the set of all video frames.
Output: $G$, the background image
Set the segmentation result of keyboard as k; Set the segmentation result of hand as h
and other regions as bh; Set the initial judgment flag as true;
for $f \in F$ do
    k ← keyboard segmentation f;
    if k == NULL then
        continue;
    else if k !=NULL and flag == true then
        G = f;
        flag = false;
    end if
    h, bh ← hand segmentation f;
    if h == NULL then
        $G$ = f;
        break;
    else
        $G$[bh] = f[bh];
    end if
    b, bh ← hand segmentation G;
    if h == NULL then
        break;
    end if
end for
return $G$
```

The background image is an image that contains only the keyboard and no hands. Most of the previous methods do not have independent hand detector, so which need to manually select the background image. We propose an automatic background selection algorithm by combining the hand and piano keyboard segmentation results. The pseudo code is shown in Algorithm 1.

We use the same method for key detection as in [5]. In the background image, we use adaptive threshold to distinguish white keys from black keys, and detect the position of the black keys by binarization and connected domain detection algorithms. According to the standard size of the piano keys and the fixed position distribution of the white keys and black keys, the position of the white keys can be estimated from the black keys. Finally we get the set of positions of the black keys and white keys. Each key is identified by its bounding rectangle. Let $B = \{k_1^{black} \dots . k_n^{black}\}$ and $W = \{k_1^{white} \dots . k_m^{white}\}$ be the set of localized black and white keys.

3.2 Hand Detection

Hand detection has two functions in our system. One is for automatic background update, which updates the area without hands in the video frame to the background image, and the other is is to limit area for pitch detection. We also use the PSPNet to accurately detect hand. In order to improve the detection efficiency, we use a lightweight network model. Semantic segmentation is a pixel-level detection method. We can know the specific position of the hand in the keyboard and remove false pitches outside the range of fingers. For example, if the hand is placed on the white key, but the adjacent black key is detected, it means that black key is false pitch.

3.3 Dynamic Background Update

Our system uses difference image for music transcription. In order to adapt lighting's changes and highlight changes caused by pressed key, we propose a new dynamic background update algorithm. The updated background at time t is B^t, is defined by

$$B^t = \begin{cases} F_{i,j}^{t-1}, & i \in H^{t-1} \\ B_{i,j}^{t-1}, & otherwise \end{cases} \qquad (1)$$

$B_{i,j}^{t-1}$ is the background image at time *t*-1 and $F_{i,j}^{t-1}$ is the video frame at time *t*-1 and H^{t-1} is hand position coordinates of $F_{i,j}^{t-1}$. In other words, the background image at time *t* is related to the background image and the video frame at time *t*-1. And the pixels outside the hand area are replaced with pixels corresponding to the video frame at time *t*-1. The area covered by the hand is the same as the background image at time *t*-1. This algorithm guarantees that there is a difference between the background image and the video frame in the hand area, and the remaining area has the smallest difference.

3.4 Feature Vectors Extraction and Pitches Detection

After obtaining the key location and the difference image, in order to detect the pressed keys in a frame, we consider the keys with overlapping areas of the hands as candidates for generating feature vectors. Each key is identified by the upper-left and lower-right corner coordinates of the bounding rectangle. For all samples, the bounding rectangle corresponding to each black and white key is resized to a fixed size of 112x32.

In the design of the model structure, in order to fully extract the characteristics of the input data and enhance the recognition ability of the model, we choose a 4-layer convolution, and add a batch-norm[17] layer after the convolution layer to accelerate the model's convergence and improve model generalization ability and prevent overfitting. The model structure is shown in Figure 2. In the design of the loss function, considering the imbalance of the positive and negative samples of the training data, the positive samples are weighted. The formula is as follows.

$$l = \sum -l(i) \log p(i) * alpha - (1 - l(i)) \log(1 - p(i)) \qquad (2)$$

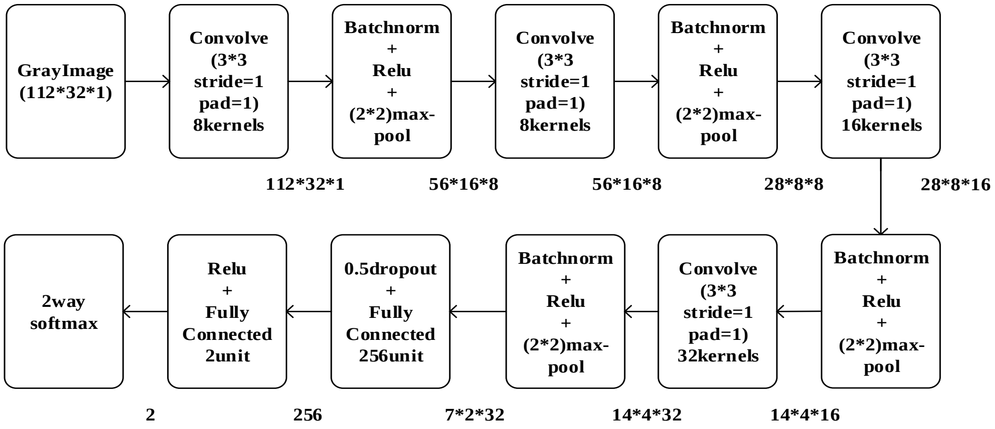
GrayImage (112*32*1)
Convolve (3*3 stride=1 pad=1) 8kernels
Batchnorm + Relu + (2*2)max-pool
Convolve (3*3 stride=1 pad=1) 8kernels
Batchnorm + Relu + (2*2)max-pool
Convolve (3*3 stride=1 pad=1) 16kernels
112*32*1
56*16*8
56*16*8
28*8*8
28*8*16
Batchnorm + Relu + (2*2)max-pool
Convolve (3*3 stride=1 pad=1) 32kernels
Batchnorm + Relu + (2*2)max-pool
0.5dropout + Fully Connected 256unit
Relu + Fully Connected 2unit
2way softmax
14*4*16
14*4*32
7*2*32
256
2

Figure 2 CNN architecture used in this work

l represents the label of the input data, and *p* represents the probability of the model output, *alpha* is the weighted value of the positive samples, used to reduce the imbalance of the positive and negative samples.

4. EXPERIMENT

In this section, we will experimentally demonstrate the robustness and result of our proposed method on various datasets, as well as the datasets and metrics for evaluation.

4.1 Dataset

Three types of datasets are mainly used in our system, the piano keyboard segmentation dataset, the hand segmentation dataset, and the piano transcription dataset.

The piano keyboard segmentation dataset includes 9443 images. We label each image with labelme[18] software, 7554 images are used as train set and 1889 images are used as test set.

For the hand segmentation dataset, we use the EgoHands and EgoYouTubeHands datasets. We divide the train set and test set according to the ratio of 80% and 20%. The final train set has 5441 images and the test set has 1062 images.

The piano transcription dataset consists of three parts. The first is the dataset proposed in [6], we define it as PianoDataset1. The second is a high-resolution video dataset that we create containing different lighting and camera positions, defined as PianoDataset2. In order to verify the performance of our system, we download some videos of different levels played by professional pianists on the internet, which can better reflect the actual performance, defined as PianoDataset3. We use the train set of PianoDataset1 and PianoDataset2 to train and evaluate the performance of the system on this three datasets, respectively. PianoDataset2 and PianoDataset3 are made in the testing environment(Figure 3)

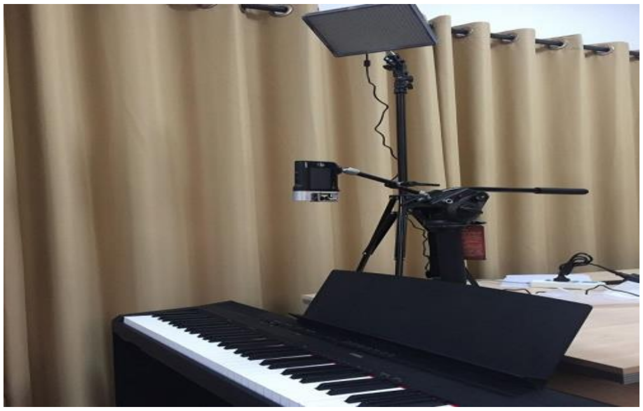

Figure 3 Testing environment

4.2 Metrics

We use frame-level metrics to assess the performance of the proposed system. The precision, recall and F1 Score are both used for frame-level evaluation. And the metrics are defined as:

$$P = \frac{N_{TP}}{N_{TP} + N_{FP}} \tag{3}$$

$$R = \frac{N_{TP}}{N_{TP} + N_{FN}} \tag{4}$$

$$F1 = \frac{2 * P * R}{P + R} \tag{5}$$

In Equation 7, 8 and 9, *P* is precision, *R* is recall and *F1* is the F1 Score which is a comprehensive score that considers the precision and recall. And N_{TP} is the number of true positives, N_{FP} is the number of false positives and N_{FN} is the number of false negatives.

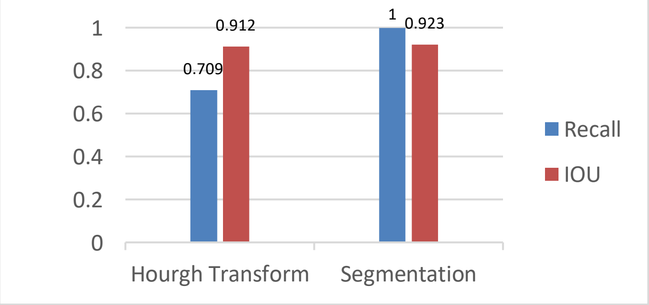

Figure 4 Comparison of hough transform and semantic segmentation on keyboard detection

4.3 Result

4.3.1 Keyboard Segmentation Result

We use semantic segmentation to detect piano keyboard, MIoU is 0.95. In order to compare the accuracy and robustness of the method based on semantic segmentation, 220 images are selected from different lighting, different shooting angles and different backgrounds, which are detected by hough transform and semantic segmentation respectively. The result is shown in Figure 4. It can be seen that the recall of hough transform is 70.9%, and the recall of semantic segmentation is 100%. The IoU of semantic segmentation is 1% higher than hough transform, which makes detection more accurate. Thus, our method enables a more robust transcription system.

4.3.2 Different Environment Result

Vision-based piano transcription system is affected by factors such as light intensity, light position and camera position. In order to prove the robustness of our system and better deployment of the system in actual scenarios, we experimentally compare the effects of light intensity, light position and camera position on accuracy, and propose the best deployment environment for the system.

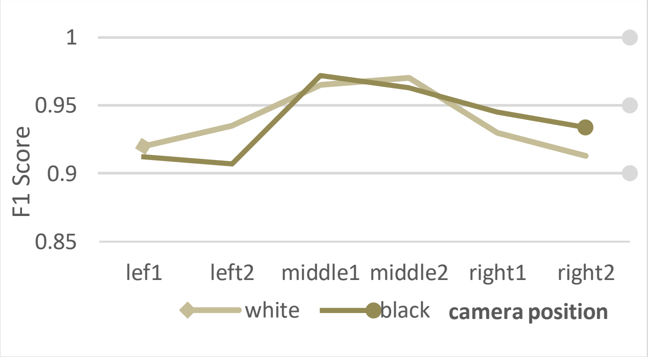

Figure 5 effect of camera position on system performance

We put the camera position on the left, middle and right of the piano without adding extra light source, and record two musics at each position to ensure that the musics are the same. The result is shown in Figure 5, it can be seen that the best result is in the middle and the F1 Score is above 0.95. Due to the angle of view of the two sides of the camera, the shadow of the keys on the other side will not change significantly, resulting in slightly lower performance, but the F1 Score is still above 0.9.

Then we fix the camera position in the middle of the piano, add extra light source and place it on the left, middle, and right of the piano. two musics are recorded at each position to ensure that the musics are the same. The result is shown in Figure 6. It can be seen that the light position has the best performance in the middle of the piano, the F1 Score is above 0.95. But the F1 Score is also above 0.9 on the left and right.

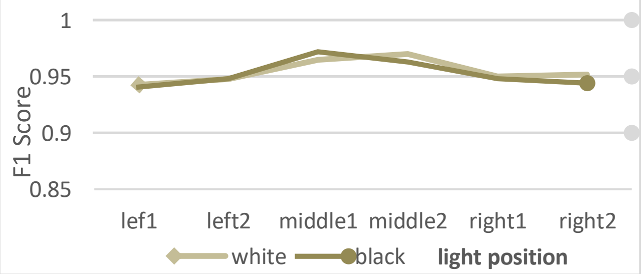

Figure 6 effect of light position on system performance

Finally, we place the camera and light in the middle of the piano, adjust the brightness of the light and measure it with a brightness meter. The brightness is [100, 200, 300, 400, 500, 600, 700, 800, 900] and ensure that the music at each brightness is same. We do not consider brightness below 100 and above 900. The result is shown in Figure 7. The performance difference is not significant under these nine light intensities, and the F1 Score is all above 0.94, which proves that our system is not sensitive to light intensity.

Based on the above experiments ,we conclude that the camera and the light source is in the middle of the piano, the system has the highest performance. The light intensity has little effect on the performance. F1 Scores are all above 0.9, which proves the robustness of our system.

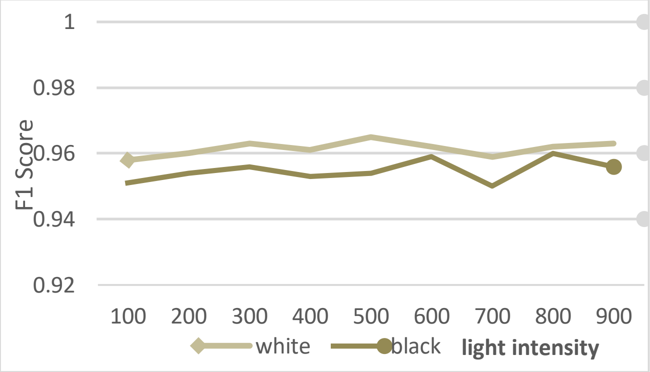

Figure 7 effect of light intensity on system performance

4.4.3 System Result

We evaluate our system on the PianoDataset1 which has 10 videos with different speeds, complexitys, and camera positions. The CNN-SVM classifier in [6] achieves an F1 Score of 0.95, and our system achieves higher performance. Table1 shows the classification results of white and black keys according to the F1 Score evaluation.

As shown in the Table1, our system achieves the highest accuracy with F1 score of 0.97 for the black keys and 0.96 for the white keys. Although the CNN-SVM classifier proposed by [6] also achieves F1 score of 0.95, it is not an end-to-end model. Furthermore, the CNN model designed by [6] is relatively simple, so the CNN[6] has insufficient fitting ability and relatively low classification accuracy. Our system is 4.5% higher than the CNN [6]. As mentioned in [6], the dynamic background update greatly helps improve the final result, so the above methods are tested using the dynamic background update.

Table1 classification results of white and black keys on the PianoDataset1

video	Ours		CNN-SVM[6]		CNN[6]	
	B	W	B	W	B	W
V1	**0.91**	0.95	0.88	**0.97**	0.87	0.88
V2	**0.98**	0.96	**0.91**	0.95	0.89	0.83
V3	**0.98**	0.97	**0.95**	0.95	0.97	0.93
V4	**0.99**	0.97	**0.97**	0.97	0.92	0.94
V5	**0.98**	0.96	0.96	**0.98**	0.92	0.94
V6	**0.98**	0.97	0.96	0.96	0.91	**0.98**
V7	**0.99**	0.93	**0.97**	0.91	0.95	0.88
V8	**0.98**	0.94	**0.97**	0.93	0.94	0.87
V9	**0.98**	0.96	**0.94**	0.96	0.93	0.95
V10	**0.97**	0.97	**0.94**	0.96	0.97	0.92
Average	**0.97**	0.96	**0.94**	0.96	0.93	0.91

Due to the low complexity of PianoDataset1's test set, we further evaluate the performance on the PianoDataset2. At the same time, we reproduce the CNN model proposed in [6]. The result is shown in Table2.

Table2 classification results of white and black keys on the PianoDataset2, P represents precision, R represents recall.

	Black Key			White Key		
	P	R	F1	P	R	F1
Ours	**0.945**	**0.937**	**0.94**	**0.964**	**0.973**	**0.968**
CNN[6]	0.882	0.896	0.889	0.905	0.914	0.909

It can be seen from the Table2 that our method has a black key F1 Score of 0.94 and a white key F1 Score of 0.968 on the PianoDataset2, which is higher than the CNN model proposed by [6].

Table3 classification results of white and black keys on the PianoDataset3, P represents precision, R represents recall.

	Black Key			White Key		
	P	R	F1	P	R	F1
Ours	**0.927**	**0.915**	**0.921**	**0.943**	**0.951**	**0.947**
CNN[6]	0.874	0.871	0.872	0.894	0.907	0.9

In order to verify the generalization ability of our system, we evaluate it on the PianoDataset3. The result is shown in Table3. As shown in the table 3, our method also achieve an average F1 Score of 0.93 in videos played by professional pianists, which is more accurate than the CNN model proposed by [6]. This proves that our method can achieve good performance in different musics with strong robustness and generalization ability.

5. CONCLUSIONS

In this paper, we improve the various stages of the previous vision-based piano transcription system and propose a more robust visual piano transcription system. Our system uses semantic segmentation to detect piano keyboard and pianist's hands. This can solve the strict requirements of the hough transform on the environment and allow our system to work stably. At the same time, a CNN model with stronger classification performance is designed. By adjusting the reasonable CNN input size, reasonable model layers and positive sample weights, the classification ability of pressed keys is improved. And in view of the lack of dataset in the current field of visual piano transcription, we propose a dataset recorded in a real scene. Our system significantly outperforms the state-of-the-art approaches on the published dataset and the dataset we proposed.

The vision-based piano transcription system proposed in this paper has achieved good results in the environment where the camera is completely fixed and the light is constant, but there are certain limitations, such as changing light intensity and moving camera and piano. Some of these limitations are inherent and can't be eliminated, but some can be solved, such as adding piano keyboard tracking. At the same time, the analysis of piano music includes not only multi-pitch transcription, but also fingering exercise and so on. In future work, we need to improve the limitations of the system as much as possible, enrich dataset and add more vision-based piano music algorithms.

6. ACKNOWLEDGEMENTS

This work is supported by The National Natural Science Foundation of China (No. 61877060).

7. REFERENCES

[1] Benetos, Emmanouil, et al. "Automatic music transcription: challenges and future directions." Journal of Intelligent Information Systems 41.3 (2013): 407-434.

[2] Cheng, Tian, et al. "An attack/decay model for piano transcription." ISMIR, 2016.

[3] Suteparuk, Potcharapol. "Detection of piano keys pressed in video." Dept. of Comput. Sci., Stanford Univ., Stanford, CA, USA, Tech. Rep (2014).

[4] Akbari, Mohammad, and Howard Cheng. "Clavision: visual automatic piano music transcription." NIME. 2015.

[5] Akbari, Mohammad, and Howard Cheng. "Real-time piano music transcription based on computer vision." IEEE Transactions on Multimedia 17.12 (2015): 2113-2121.

[6] Akbari, Mohammad, Jie Liang, and Howard Cheng. "A real-time system for online learning-based visual transcription of piano music." Multimedia Tools and Applications 77.19 (2018): 25513-25535.

[7] Moorer, James A. "On the transcription of musical sound by computer." Computer Music Journal (1977): 32-38.

[8] Goodwin, Adam, and Richard Green. "Key detection for a virtual piano teacher." 2013 28th International Conference on Image and Vision Computing New Zealand (IVCNZ 2013). IEEE, 2013.

[9] Vishal, Boga, and K. Deepak Lawrence. "Paper piano—Shadow analysis based touch interaction." 2017 2nd International Conference on Man and Machine Interfacing (MAMI). IEEE, 2017.

[10] Kang S, Kim J, Yoon S. Virtual Piano using Computer Vision[J]. arXiv preprint arXiv:1910.12539, 2019.

[11] Frisson, Christian, et al. "Multimodal guitar: Performance toolbox and study workbench." QPSR of the numediart research program. Ed. by Thierry Dutoit and Beno î Macq 2 (2009): 3.

[12] Paleari, Marco, et al. "A multimodal approach to music transcription." 2008 15th IEEE International Conference on Image Processing. IEEE, 2008.

[13] Wan, Yu Long, et al. "Automatic transcription of piano music using audio-vision fusion." Applied Mechanics and Materials. Vol. 333. Trans Tech Publications, 2013.

[14] Wan, Yulong, et al. "Automatic Piano Music Transcription Using Audio-Visual Features." Chinese Journal of Electronics24.3 (2015): 596-603.

[15] Lee, Jangwon, et al. "Observing Pianist Accuracy and Form with Computer Vision." 2019 IEEE Winter Conference on Applications of Computer Vision (WACV). IEEE, 2019.

[16] Zhao, Hengshuang, et al. "Pyramid scene parsing network." Proceedings of the IEEE conference on computer vision and pattern recognition. 2017.

[17] Szegedy, Christian, et al. "Rethinking the inception architecture for computer vision." Proceedings of the IEEE conference on computer vision and pattern recognition. 2016.

[18] Wada K. labelme: Image Polygonal Annotation with Python[J]. 2016.