参赛队员姓名：Jasmine Liu

中学：Shanghai American School

省份：上海

国家/地区：中国南部赛区(上海)

指导教师姓名：陈文光，刘长松

指导教师单位：清华大学计算机系/电子工程系

论文题目：An Assistance System for the Visually Impaired with Scene Curved Text Detection and Recognition

An Assistance System for the Visually Impaired with Scene Curved Text Detection and Recognition

Jasmine Liu*
jasmine02pd2024@saschina.org
Shanghai American School
Shanghai, China

Abstract

Reading package labels at the grocery store, following signs at the airport to the correct gate, and checking the names of restaurants along the street are all activities that visually impaired people find hard to perform in their daily lives. Although existing OCR systems have high accuracy in recognizing organized text, they are less refined in recognizing scene text, which is often distorted and curved.

To solve these challenges, I developed an assistance system that contains 2 different scene curved text detection and recognition algorithms with different balances between accuracy and speed: the two-stage algorithm Differentiable Binarization Network (DBNet) and End-to-End algorithm Point Gathering Network (PGNet). The algorithms are trained using 3 scene curved text datasets and then deployed to a server. I also developed a user-friendly mobile APP to call on these algorithms, along with other types of recognition. To help visually impaired people use my system, I added speech recognition and speech synthesis for APP control. The system is combined with a camera attached to a pair of ordinary glasses, allowing users to recognize scene curved text and objects in various situations everyday.

In the experiment, the system using the optimized DBNet with my novel text rectification module achieves a 92% recognition accuracy for scene curved text, around 8% higher than popular general text recognition systems. The system with the optimized End-to-End PGNet and OpenVINO achieves the highest efficiency at around 2 seconds per image and a 90% accuracy. I received positive feedback from trials with visually impaired users and ophthalmologists.

Here is the project source code:
https://github.com/jasmine6524/Scene-Curved-Text-Recognition-System

1 Introduction

Every day, the vast majority of tasks we perform require our comprehension of text. In 2015, there was an estimated number of 253 million people who suffer from visual impairment on a global scale [1]. There are various forms of visual impairment, such as low vision (between 0.1 and 0.3), loss of central vision, peripheral vision, diplopia, and many more [2]. Currently, visually impaired people face difficulties when navigating our society due to their inability to recognize scene text because they are often curved and distorted. This difficulty is evident in their lives when shopping at grocery stores, understanding restaurant labels, interpreting direction and warning signs, etc. (Figure 1). Unfortunately, traditional text recognition solutions cannot address this issue. It is inevitable for visually impaired people to have a solution that can help them overcome this barrier to improve their living standards and become more independent in the future.

Figure 1: Examples of Scene Curved Text [12]

Starting in 2020, I joined a community service organization called ORBIS where I teach Chinese visually impaired children English. During these experiences, I interacted with visually impaired children and developed friendships with them. Through these interactions, I have heard about numerous difficulties in their daily lives, with recognizing scene text everyday as the main concern. This prompted my aspiration to reduce the amount of inconvenience that visually impaired people face when traveling around our society in their daily lives. After communicating with visually impaired people to learn about their needs for recognizing scene text, I researched about existing solutions to determine whether they can adequately address their difficulties.

Existing solutions of scene curved text recognition for the visually impaired are either magnification products or Optical Character Recognition (OCR) systems and smart glasses.

- `Clinical Solutions`: Dr. Xue, an ophthalmologist from Fudan University's Affiliated Hospital, said the most popular clinical solutions that visually impaired people use to recognize text are magnifying glasses. However, they are often inconvenient to use when navigating in outdoor environments.
- `OCRs and Smart Glasses`: Although OCRs can accurately recognize organized text, they are inaccurate with scene curved text because they have not been optimized for these situations. Among the minimal options of smart glasses that have been developed for the visually impaired, visually impaired citizens are often disincentivized to purchase these products due to the high price, with Orcam [3] specifically priced at around $3000 to $5000.

Therefore, current solutions cannot address the challenges that visually impaired people encounter because they are either inconvenient, inaccurate, or not affordable.

To address their difficulties, this paper develops an AI system that can help visually impaired people recognize scene curved text accurately and conveniently in their daily lives. Two different

text detection and recognition algorithms are trained to provide visually impaired users with different balances between accuracy and speed. The scene curved text recognition accuracy and response time match their needs: the algorithm with a 90% accuracy has an optimized 2 seconds response time for outdoor usage while the other algorithm has a 92% accuracy with a 17 seconds response time for usage at home. To help visually impaired people use this system, I developed a mobile APP with speech recognition for APP control and speech synthesis to read aloud the recognized results. Other categories of recognition are also included, such as currency, object, and face recognition.

This system also addresses the issue of high costs with existing smart glasses. They are often not cost-effective for the visually impaired because these products solely rely on the computing power of embedded components integrated into the glasses. To decrease the cost, I leveraged cloud computing power and phone computing power.

To summarize, this paper's main contributions are:

1.Developed an assistance system based on cloud-client architecture. This system contains a user-friendly mobile APP in the client end and recognition algorithms in the server end (Figure 4). The cost is reduced by leveraging cloud computing power and phone computing power.

2. Created a text trimming and rectification module to improve accuracy. I created a novel text rectification module and inserted it between the detection and recognition modules to improve two-stage recognition accuracy. The rectification module is capable of cropping out each text instance and rectifying the curved texts into organized texts.

3. Leveraged OpenVINO framework to increase speed. To offer a fast response time for outdoor usage, I optimized the End-to-End PGNet [22] algorithm with OpenVINO [27]; this further increased PGNet's inference speed to 2 seconds per image.

2 Related Work

2.1 Clinical Solutions

To understand the clinical solutions that visually impaired people most often use for scene text recognition everyday, I communicated with Dr. Xue, an ophthalmologist from Fudan University's Affiliated Hospital. She stated that popular text recognition solutions for her visually impaired patients are based on magnification (Figure 2).

Figure 2: Clinical Solutions [4]

These magnifying solutions for the visually impaired can be split into two main categories: non-optical and optical visual aids. Non-optical aids do not use magnifying lenses to assist people's vision. These include screen readers, smart projectors through Bluetooth, and video magnifiers. On the other hand, optical visual aids do use magnifying lenses to improve visual performance. These include handheld magnifiers, stand magnifiers, and telescopes, which often minimize users' visual field [4]. Although these solutions may be effective in stationary scenarios, they are neither very convenient nor accurate when visually impaired people need to recognize scene curved text outdoors in our society. Curved and distorted text will remain curved and distorted even when magnified. Consequently, these common and accessible traditional solutions are inadequate for addressing visually impaired people's issue with scene curved text recognition.

2.2 OCRs and Smart Glasses

Aside from clinical solutions, recent technological advancements have also produced Optical Character Recognition (OCR) systems and smart glasses that are available for visually impaired people to use.

2.2.1 *OCRs*

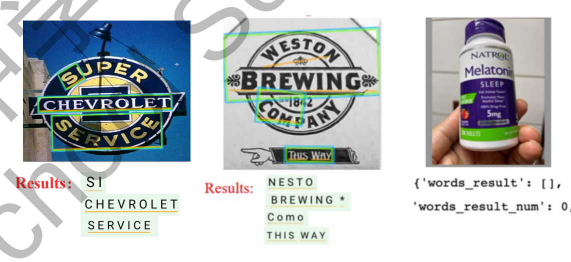


Figure 3: Optical Character Recognition Systems

After testing popular OCRs, the results showed that they are accurate in recognizing organized text. However, the different angles, backgrounds, and shapes of scene texts significantly decrease the recognition accuracy of OCRs (Figure 3). This is because these popular OCRs are general text recognition systems that specialize in recognizing horizontal, organized text. Meanwhile, the most challenging types of text that visually impaired people need to recognize everyday are often scene curved text. Hence, current general text recognition systems are not fully capable of helping visually impaired people.

2.2.2 *Smart Glasses*

Even though there are a few products that integrate cameras and glasses that are convenient and accurate for text recognition, they are not optimized for scene curved text and the options are minimal with the prices being extremely expensive. To the best of our knowledge, there are two companies that have created similar products of smart glasses for the visually impaired: Orcam and AngelEye. Although Orcam has a variety of functions including object and text recognition, it is priced at around $3000 to $5000. For AngelEye, they have not updated their smart glasses recently [5]. So, there are not many accessible and affordable smart glasses for the visually impaired to use in their daily lives.

Evidently, most of the current solutions are either inaccurate, inconvenient or not affordable for visually impaired people to use everyday.

3 Methodology

3.1 System Design

3.1.1 *Requirements*

Prior to developing my system, I communicated with my visually impaired students to better understand their needs for recognition.

- `Outdoor usage`: They would prefer a recognition system with a faster response time. They hope the results can be returned within 5 seconds and the accuracy to be above 85%.
- `Indoor usage at home`: They prefer to have a higher accuracy, above 90%, so they are willing to tolerate up to 20 seconds to hear the accurate results.

3.1.2 *System Architecture*

To address visually impaired people's needs, I developed an assistance system based on cloud-client architecture. There is a user-friendly mobile APP on the client side and scene curved text recognition algorithms on the server side (Figure 4). The scene curved text recognition algorithms have a high accuracy and fast response time that meets their needs. Other categories of recognition, including object, currency, and face recognition are also included on the server end.

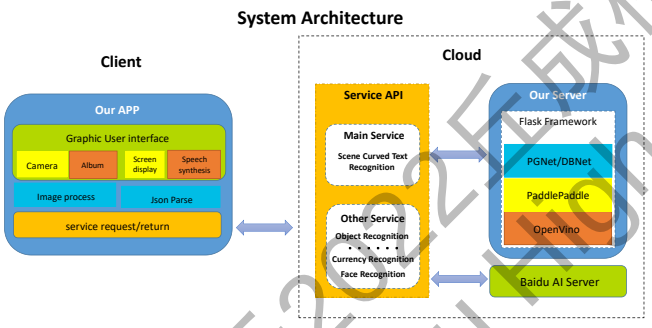


Figure 4: System Architecture

In the client end, the mobile APP obtains the image and performs image-processing before sending the image to cloud for recognition. The recognized results are returned to the client end and go through json parse. Finally, they are displayed on the APP screen and speech synthesis reads aloud the results.

In the cloud end, the service API consists of my main service, the scene curved text recognition, and other services, including object, currency, and face recognition. For the main service, the PGNet algorithm is optimized using OpenVINO and implemented onto a Flask Web Framework. Other services call on Baidu's AI server (Figure 4).

The complete vision assistance system for scene curved text recognition consists of six stages (Figure 5): (1) Speech recognition for voice control (2) Capture an image through the mobile APP (3) Image uploaded to cloud for recognition (4) Text Detection and Recognition (5) Return recognition results to the mobile APP (6) Speech synthesis reads aloud the recognized results.

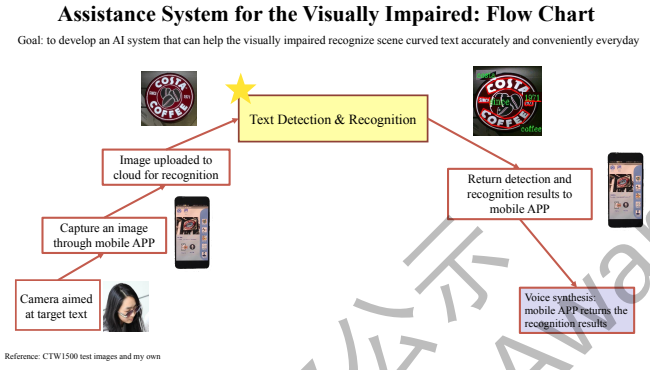


Figure 5: Vision Assistance System Flowchart

3.2 Text Detection and Recognition Algorithms

Since popular general text recognition systems cannot accurately recognize scene curved text, I researched about current text recognition algorithms based on Deep Learning that have made progress in scene curved text recognition. These text detection and recognition models can be split into two main categories: text detection plus text recognition (two-stage) and End-to-End text recognition (one-stage).

3.2.1 *Two-Stage Text Recognition Algorithm*

Two-stage detection and recognition algorithms have a greater potential in achieving high accuracy, but sacrifice some speed. This allows it to be a suitable option for fulfilling the needs of visually impaired people when they are at home. In this system, DBNet is selected as the scene curved text detection algorithm for its State-of-the-Art performance. To further improve the two-stage accuracy, I developed a novel text rectification module and added it in between DBNet text detection and text recognition. This provides visually impaired users with a high accuracy of 92% and a response time within 20 seconds.

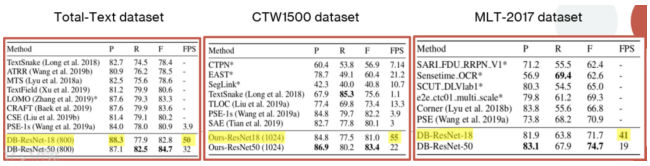
Why DBNet?

Semantic segmentation and object detection are the two main types of text detection algorithms.

- `Semantic Segmentation Algorithms`: Semantic segmentation algorithms determine each pixel's probability of being a text area, producing a binary map that separates text from non-text regions. Examples of semantic segmentation models include PSENet [6] and Mask TextSpotter [7].
- `Object Detection Algorithms`: Object detection algorithms have a Region Proposal Network [9] that produces different anchors, or different sizes and proportions of boxes, that go through selection and box regression, aiming for the ground truth. Object detection models include Fast R-CNN [8] Faster R-CNN [9], Mask R-CNN [10] and SPCNet [11].

Since semantic segmentation algorithms analyze images on a more precise level, they possess an advantage in recognizing smaller

targets, and scene curved texts in our environment often appear as small targets in a background. Within this category, the different algorithms were compared on five benchmark datasets, including CTW1500 [12], Total-Text [13], and MLT-2017 [14]. These datasets consist of curved, multi-oriented, and horizontal text images. The two most accurate and advanced semantic segmentation models are DBNet and PSENet. DBNet has a comparatively higher speed in the inference stage because the post-processing after kernel generation in PSENet is fairly time-consuming; DBNet also has the highest accuracy on several benchmark datasets (Figure 6) [15].



Total-Text dataset

Method	P	R	F	FPS
TextSnake (Long et al. 2018)	82.7	74.5	78.4	-
ATRR (Wang et al. 2019b)	80.9	76.2	78.5	-
MTS (Lyu et al. 2018a)	82.5	75.6	78.6	-
TextField (Xu et al. 2019)	81.2	79.9	80.6	-
LOMO (Zhang et al. 2019)*	87.6	79.3	83.3	-
CRAFT (Baek et al. 2019)	87.6	79.9	83.6	-
CSE (Liu et al. 2019b)	81.4	79.1	80.2	-
PSE-1s (Wang et al. 2019a)	84.0	78.0	80.9	3.9
DB-ResNet-18 (800)	88.3	77.9	82.8	50
DB-ResNet-50 (800)	87.1	**82.5**	**84.7**	32

CTW1500 dataset

Method	P	R	F	FPS
CTPN*	60.4	53.8	56.9	7.14
EAST*	78.7	49.1	60.4	21.2
SegLink*	42.3	40.0	40.8	10.7
TextSnake (Long et al. 2018)	67.9	**85.3**	75.6	1.1
TLOC (Liu et al. 2019a)	77.4	69.8	73.4	13.3
PSE-1s (Wang et al. 2019a)	84.8	79.7	82.2	3.9
SAE (Tian et al. 2019)	82.7	77.8	80.1	3
Ours-ResNet18 (1024)	84.8	77.5	81.0	55
Ours-ResNet50 (1024)	**86.9**	80.2	**83.4**	22

MLT-2017 dataset

Method	P	R	F	FPS
SARI_FDU_RRPN_V1*	71.2	55.5	62.4	-
Sensetime_OCR*	56.9	**69.4**	62.6	-
SCUT_DLVlab1*	80.3	54.5	65.0	-
e2e_ctc01_multi_scale*	79.8	61.2	69.3	-
Corner (Lyu et al. 2018b)	83.8	55.6	66.8	-
PSE (Wang et al. 2019a)	73.8	68.2	70.9	-
DB-ResNet-18	81.9	63.8	71.7	41
DB-ResNet-50	**83.1**	67.9	**74.7**	19

Figure 6: Two-Stage Algorithms Comparison [15]

DBNet's proposed Differentiable Binarization process allows DBNet to provide a robust binarization map while being faster than past leading methods. Therefore, DBNet is the most suitable algorithm for this project as the State-of-the-Art semantic segmentation text detection algorithm. For the backbone, ResNet-18 was selected to fulfill visually impaired people's needs of a response time within 20 seconds for recognition at home. ResNet-18 balances accuracy and speed, with the accuracy being fairly similar to that of ResNet-50, but the speed (FPS) doubled [16].

Differentiable Binarization:

DBNet is unique for its main contribution of Differentiable Binarization.

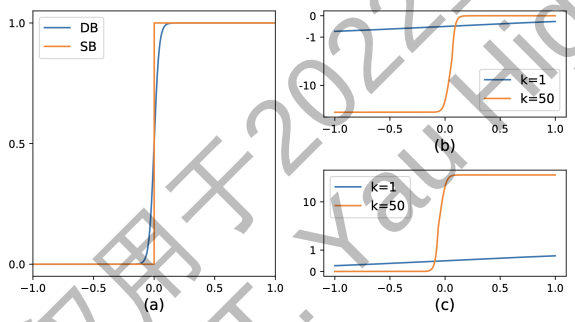


Figure 7: Standard Binarization vs. Differentiable Binarization [15]

Here is a graphical comparison between Standard Binarization and Differentiable Binarization (Figure 7).

$$B_{i,j} = \begin{cases} 1 & \text{if } P_{i,j} >= t, \\ 0 & \text{otherwise.} \end{cases} \quad (1)$$

Previous methods using Standard Binarization set a fixed threshold for each pixel, so the threshold values are unable to be differentiated (Equation 1 [15]).

$$\hat{B}_{i,j} = \frac{1}{1 + e^{-k(P_{i,j} - T_{i,j})}} \quad (2)$$

DBNet proposes Differentiable Binarization for joint optimization. When training DBNet, variables go through forward and backward propagation because they are derivable. Instead of adopting standard binarization methods that have a non-differentiable function, DBNet slightly modified it into an approximate step function that can be differentiated and derivable (Equation 2 [15]). This Differentiable binarization allows the threshold of each pixel to be different and derivable. This means that the threshold of each pixel can also experience forward and backward propagation. Since each propagation results in an adjustment of the variable, the Differentiable Binarization process can help DBNet create a more accurate binary map than other text detection algorithms.

Overall Structure:

The overall structure of DBNet consists of two stages: the Feature Pyramid Network (FPN) and post-processing. The output of FPN is a feature map that is processed to create a probability map and a threshold map (Figure 8).

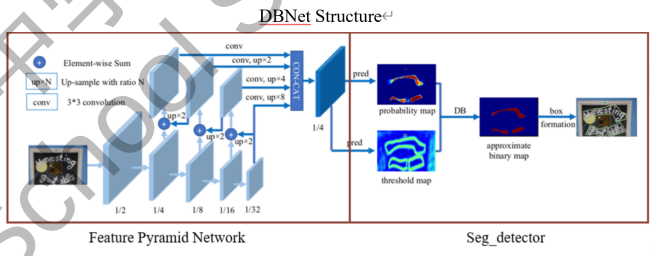


Figure 8: Differentiable Binary Net Structure (DBNet) [15]

The Feature Pyramid Network (FPN) section mainly consists of ResNet-18 and the DB Head. FPNs combine low-level and high-level feature maps to recognize targets of different sizes. Low-level feature maps consist of small features that have precise location information; high-level feature maps consist of large features and semantic information [17, 35].

In the ResNet-18 backbone, there are five convolution stages that shrink the original image through the usage of 3x3 kernels and deformable convolutions [31]. I chose to use deformable convolutions because they possess the advantage of covering a larger area that allows for more accurate identification of key characteristics.

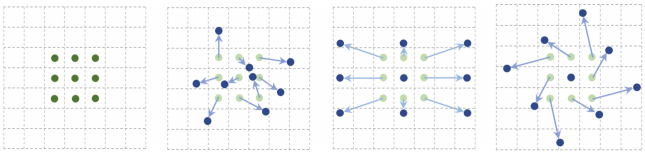


Figure 9: Deformable Convolution Window [18]

The purpose of Deformable Convolutions is to accurately identify and focus the convolution process with kernels on the target area, which is achieved through the deformation of sampling locations.

This deformation is done by adding offsets to regular sampling locations in a standard convolution, resulting in the free deformation of sampling grids (Figure 9).

Figure 10: Deformable Convolution Visual Depiction [18]

Thus, their advantage is to cover a larger area, allowing for more accurate identification of key characteristics (Figure 10).

The output feature maps of each convolution layer will go through convolution and then a series of functions, including the up-sampling of each feature map by a scale factor of two. This makes adding feature maps possible because their depths and sizes are now equal. The result of adding the feature maps is a "fuse" map. The advantage of FPN is that it retains the features from all the convolution layers (Figure 10).

The post-processing section of DBNet consists of the probability map, threshold map, and differentiable binary map. Firstly, the "fuse" map is used to create a probability map of the original picture. After the probability map is created, it goes through up-sampling to create a new probability map, which is then combined with the original "fuse" map to create a new "fuse" map. The new '"fuse" map is then used to create a threshold map. Afterwards, the probability and threshold maps are altered to create the approximate binary map, which is differentiable.

3.2.2 *Novel Text Rectification Module*

Following DBNet's text detection algorithm, the curved text needs to be rectified into a normal text for the recognition module to recognize it more easily. This allows the two-stage algorithm to obtain a higher scene curved text recognition accuracy. Hence, I developed this novel text rectification module and inserted it after DBNet's text detection and before text recognition (Figure 11).

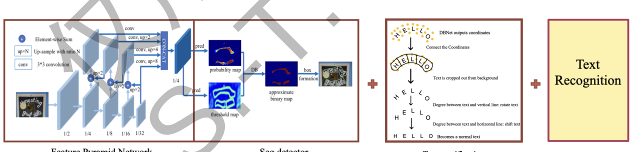


Figure 11: Two-Stage Algorithm Flowchart

Current text rectification methods are for normal, organized text, performed through rectangular and elliptical bounding boxes. However, these methods are unable to rectify curved texts to improve recognition accuracy (Figure 12). Hence, irregular bounding boxes are needed to rectify scene curved text [32, 33].

Figure 12: Text Rectification Issue [12]

The text rectification method for scene curved text that I propose is through rotating and rectifying each character after connecting the coordinates outputted by DBNet (Figure 13). The outputs of a text detection algorithm include a picture with boxes on the detected text regions and the coordinates of the boxes. Following the text detection algorithm, a text trimming and rectification module is developed to enhance the text recognition results. Without a rectification module, it would be very hard for text recognition modules to recognize what the scene curved text reads. Hence, text trimming and rectification acts as an essential bridge between detection and recognition.

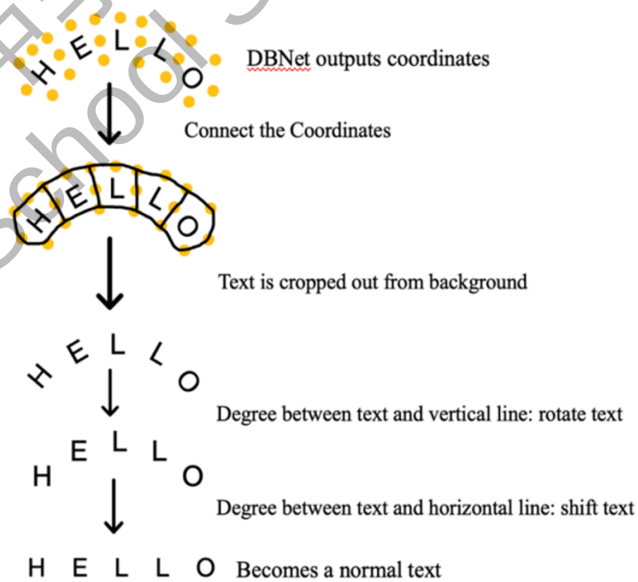


Figure 13: Text Rectification Solution

The process of trimming and rectifying text aims to crop out the text from the background scene by utilizing the coordinates file that is outputted by DBNet. These coordinates surround the outside of each text instance and in between its characters. When the coordinates are connected, the text is split into individual characters that can be separately rectified. To rectify each character, the angles between the character and the horizontal and vertical lines need to be calculated. The angle with the vertical line determines the degree of text rotation. Meanwhile, the distance of the character from the horizontal line determines how much it has to shift up or down. When these degrees are found and the corresponding

adjustments are performed, the curved text is able to become a normal text.

3.2.3 *End-to-End Text Recognition Algorithm*

End-to-End text recognition algorithms combine text detection and text recognition.

Why PGNet?

Currently, there is a minimal number of mature End-to-End models for scene curved text recognition. Among those that exist, most End-to-End models are based on two-stage frameworks or character-based frameworks, resulting in complicated and inefficient pipelines for application, including TextNet [19], ABCNet [20], and TextDragon [21]. They often rely on time-consuming operations, like character-level-annotations, Non-Maximum Suppression (NMS), and Region of Interest (RoI).

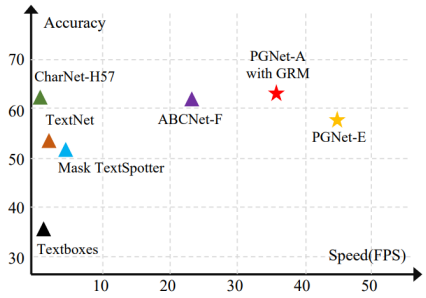


Figure 1: Model Speed vs. Recognition Accuracy on Total-Text: Our PGNet-E achieves at least two times faster than the most recent state-of-the-art method ABCNet (Liu et al. 2020) with competitive recognition accuracy. Complete results are in Table. 3.

Figure 14: End-to-End Algorithms Comparison [22]

Point Gathering Network (PGNet) is the first to propose an End-to-End text detection and recognition algorithm that does not rely on these time-consuming operations, drastically improving scene text detection and recognition speed. Compared to two-stage networks with separate text detection and recognition modules, PGNet possesses the main advantage of a faster response time. After comparing different End-to-End text recognition algorithms, PGNet exhibited state-of-the-art results. On Total Text, PGNet-Efficiency (focuses on guaranteeing high efficiency), can reach more than twice the speed of ABCNet [20], the most recent state-of-the-art method. Specifically, for curved text, PGNet's detection results are nearly 1.5% higher in accuracy than other methods. For PGNet-Accuracy (focuses on guaranteeing high accuracy), the recognition accuracy is similar to CharNet's [23], the previous most accurate network, but PGNet is nearly 30 times faster (Figure 14). Hence, PGNet was selected and optimized as the other text recognition model in my system. Specifically, PGNet-A has been chosen for its better balance between accuracy and speed. It adopts the Graph Refinement Module (GRM) [24, 25] and has a backbone of ResNet-50.

PGNet has 3 unique advantages:
1) PGNet avoids the time-consuming tasks of NMS and RoI, which are often present in 2 stage networks.
2) PGNet restores the correct reading order of text instances.
3) The Graph Refinement Module improves Connectionist Temporal Classification (CTC) [26, 34] accuracy.

Overall Structure:

The overall structure of PGNet includes the Feature Pyramid Network (FPN) and post-processing (Figure 15). The Feature Pyramid Network creates the feature map F_{visual}, which is then used to create the Text Center Line (TCL), Text Border Offset (TBO), Text Direction Offset (TDO), and Text Character Classification (TCC) maps, while the post-processing uses these maps and the Point Gathering CTC decoder to obtain the final text recognition results. The network's overall performance is improved with the Graph Refinement Module.

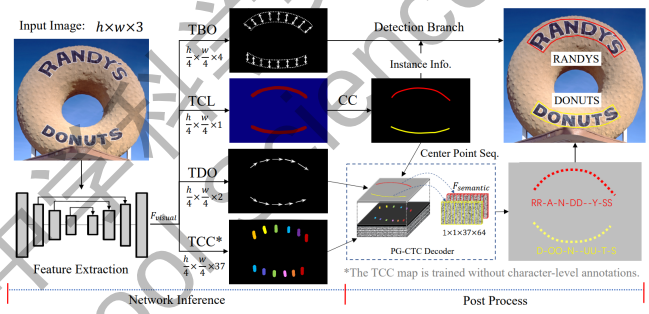


Figure 3: The pipeline of PGNet: 1) Extract feature from an input image, and learn TCL, TBO, TDO, TCC maps as a multi-task problem; 2) The detection and recognition of each text instance can be achieved in a single shot by polygon restoration and PG-CTC decoding mechanism with the center point sequence of each text region.

Figure 15: Point Gathering Net Framework [22]

Point Gathering CTC:

In the training and inference processes, Point Gathering Connectionist Temporal Classification (PG-CTC) is utilized to remove character-level annotations, NMS, and RoI to increase speed. TCC maps contain 37 different characters, including 26 letters, 10 numbers, and 1 background. Based on the center of each text instance, the character classification probabilities are gathered from TCC maps.

During the training process, previous CTC losses either suffer from background noises or cannot recognize more than 1 text instance. PG-CTC loss addresses this problem without character-level annotations, and it is capable of handling multiple text instances by incorporating transcript labels.

During inference, the PG-CTC decoder simplifies the End-to-End text recognition process. The direction of the text instances is recovered through picking out the center points, creating a sequence, and sorting it in the right reading order. The center point sequence is found through a morphological method that obtains the skeleton of a text region. The TDO provides the text direction of each point. Then, the average direction is calculated and the points are reorganized based on projection lengths to obtain the center point sequence.

For polygon restoration in text detection, the TBO map provides the corresponding border points and connects these border points to draw a complete polygon.

Label Generation:

The label generation process produces the TCL, TBO, and TDO maps. The TCL map shows the results of segregating each text instance and shrinking the boundaries. The TBO map consists of the boundary offsets for each pixel in TCL. During the inference process, the TBO map is used to detect text regions. Finally, the TDO map recovers the reading order. This is accomplished using TCL's offset vector between every pixel and the next reading point. For quadrilaterals, the offset is from the center of the left side to the center of the right side. The length of a text region, normalized by the number of characters, yields the magnitude. For multi-lateral shapes, it is viewed as multiple quadrilaterals connected together (Figure 16).

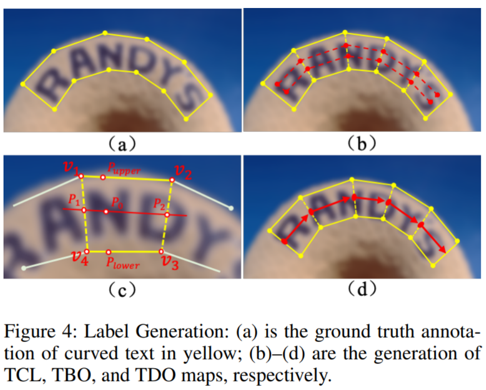


Figure 16: Label Generation [22]

Graph Refinement Module:

The Graph Refinement Module (GRM) uses word-level semantic and visual contexts to improve the network's overall performance. The inputs into this module are F_{visual} and TCC, while the output is the character classification probability sequence. There are 2 main sections, one recreates a semantic context and the other recreates a visual context (Figure 17).

For the semantic context, the input TCC goes through point gathering operations to create the F_s map. Then, it is transformed into X_s using the embed operation. Afterwards, 3 graph layers change the input X_s into Y_s.

For the visual context, the input F_{visual} goes through point gathering operations to create the F_v map. This map then goes through convolutional operations and transforms into X_v. Lastly, 3 graph layers follow to change the X_v map into Y_v.

Finally, the Y_v and Y_s maps are connected through the Fully Connected Layers (FCL) to strengthen the probability sequence, and it is modified using the CTC loss function.

3.2.4 *Optimizing PGNet's Efficiency*

To further improve PGNet's inference speed, I leveraged the OpenVINO framework. After converting PGNet from the PaddlePaddle model to the ONNX model, then to the OpenVINO IR (XML model), I successfully inserted the OpenVINO module into the inference code, creating two PGNet models that call on OpenVINO.

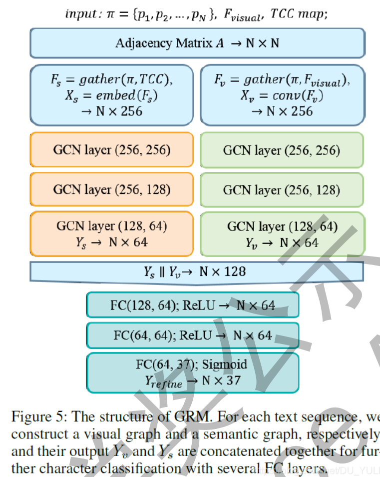


Figure 17: Graph Refinement Module [22]

Based on rounds of testing with multiple scene curved text images, the PGNet XML model showed a faster response time compared to the PGNet ONNX model. Compared to PGNet without OpenVINO, the PGNet XML model significantly increased the speed, by 3-4 times, from around 8 seconds per image to around 2 seconds. This provides visually impaired people with an option that best balances accuracy and speed for outdoor usage.

3.3 User-Friendly Mobile APP

Since traditional magnification-based solutions are inconvenient, I developed a user-friendly mobile APP to address this issue. This ensures that my system can be used by the visually impaired conveniently in their daily lives. I chose my optimized PGNet algorithm to deploy onto a server based on Flask Web Framework. Hence, my APP calls on my PGNet algorithm for scene curved text recognition by sending a service request to the server. Then, the recognition results are returned from the server and displayed on the screen (Figure 18).

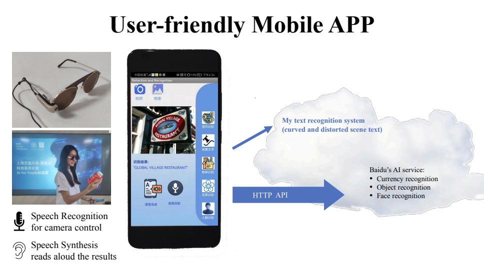
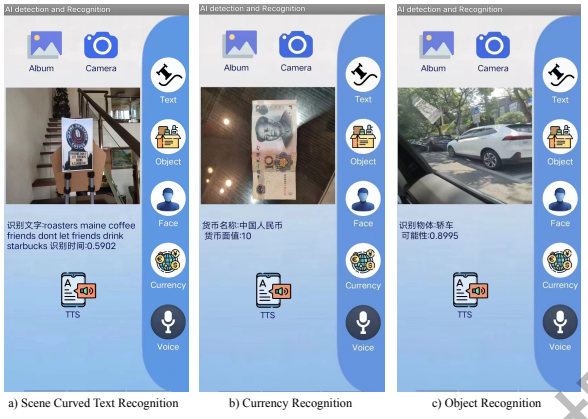


Figure 18: User-Friendly Mobile APP

After developing a prototype of this system, several visually impaired people volunteered to test it and provided some feedback. They found this system to be very helpful, and suggested that more speech modules could be implemented into the system to simplify

their usage. To help them use my mobile APP more easily, speech recognition is used for APP control. This would include using voice commands to control the camera and select a type of recognition, whether it is scene text, object, or currency recognition. Speech synthesis is used to read aloud the recognized results for visually impaired users to hear.

They also suggested me to include other types of recognition that could also assist them in their daily lives. So, I made object, currency, and face recognition available in my APP by calling on Baidu's AI service. To help users take pictures more conveniently, the camera is attached to a pair of ordinary glasses, and connected to a mobile phone (Figure 19).



a) Scene Curved Text Recognition b) Currency Recognition c) Object Recognition

Figure 19: User-friendly Mobile APP Screen

This system is more cost-effective than existing smart glasses for the visually impaired, such as Orcam and AngelEye. Smart glasses like Orcam integrate all the computing components into their glasses, where there is limited space and power consumption. Hence, the computing power is significantly restricted and the cost is very high. To operate with limited computing power, the algorithms need to be light-weight, so they cannot specialize in recognizing curved and distorted scene text. This means these smart glasses will remain highly costly while unable to effectively help visually impaired people with recognizing scene curved text. These smart glasses are also difficult to upgrade to support the latest or more complicated algorithms, such as those of action recognition, unless consumers buy new smart glasses.

On the other hand, my system leverages both cloud computing and phone computing power. Not only can this source of greater computing power allow my system to support more advanced algorithms, such as action recognition, but it can also help decrease the price. This system can also continuously upgrade without requiring consumers to buy new smart glasses, increasing the affordability of this product.

4 Experiment

4.1 Experimental Setup

4.1.1 *Selecting Datasets*

To train my scene curved text recognition algorithms, I conducted research to find suitable datasets. Among the popular datasets used to train text recognition algorithms, including MLT-2017, ICDAR-2015 [28], MSRA-TD500 [29] and CTW1500, I chose ICDAR-2015 and CTW1500 to train DBNet. This is because ICDAR-2015 contains English scene text and CTW1500 contains Chinese and English scene curved text. Each of these datasets contain 1000 training images and 500 testing images. The Total-Text dataset with scene curved text images was used to train the PGNet model; it contains 1255 training images and 300 testing images (Figure 20). This compilation of various scene text images is representative of the variations that can occur with text recognition in our daily lives.

Datasets	Size	Function	Characteristics
SynthText	800k	pre-train model: 100k iterations	synthesized from 8k background images
MLT-2017	train: 7200; val: 1800; test: 9000	Ablation study: Finetune (used train + val) - show effectiveness of DB, Dconv, and backbones - test accuracy of multi-language text detection	multi-language dataset: 9 languages (6 scripts)
ICDAR-2015	train: 1000 test: 500	Test the accuracy of multi-oriented texts	- Images captured by Google glasses: 720x1280 resolution - labeled at word level
MSRA-TD500	train: 300 test: 200 (included 400 extra training imgs from HUST-TR400)	- demonstrate the effectiveness of backbones - accuracy of multi-language text detection	- English and Chinese - labeled at text-line level
CTW1500	train: 1000 test: 500	- demonstrate the effectiveness of DB, Dconv, and backbones - test accuracy of curved text detection	- Chinese text - focuses on curved text - annotated at text-line level
Total-text	train: 1255 test: 300	- test accuracy of curved text detection	- text of various shapes (horizontal, multi-oriented, curved)

Figure 20: Scene Curved Text Datsets

4.1.2 *Model Training*

The DBNet model used for this paper was pre-trained with the SynthText [30] dataset, consisting of 80,000 scene text images. On top of that, I trained my model with ICDAR-2015 and CTW1500. When training the DBNet text detection model, a Nvidia GeForce GTX 2080 was used at first with the ICDAR-2015 dataset for 1200 epochs, which took around 70 to 80 hours. Then, a switch was made to a cloud GPU for training the DBNet model because it was faster. Using the cloud GPU, DBNet was trained with the CTW1500 dataset for 800 epochs, which took less than 30 hours. Meanwhile, the PGNet model was trained with the Total-Text dataset with English scene curved text images.

4.2 Accuracy Experiment

Results were gathered through testing the DBNet plus text rectification algorithm, the PGNet algorithm, and a general text recognition system using scene curved text pictures. Some of the pictures were scene text images we collected, others were from the CTW1500 testing dataset. The selected pictures had relatively busy backgrounds and curved texts to provide representative results about the capabilities of each text recognition system. Below are the procedures that were used for conducting the experiment:

1. Take a picture of the scene curved text image displayed on a computer screen with the camera attached to a pair of glasses, which will then be sent into the rest of our system.
2. Record the recognition accuracy of my system for that image.
3. Save the image of the scene curved text image taken by our camera on glasses, stored as "tmp.jpg".

4. Input “tmp.jpg”, the same scene curved text image that our system recognized, into the general text recognition systems.
5. Record the results of the other text recognition systems for that image.
6. Calculate the percentages for recognition accuracy through word-based recall. This means dividing the total number of words by the number of correct words, with correct words being those that are recognized without a single incorrect character.
7. Compare the percentages of these 3 systems.

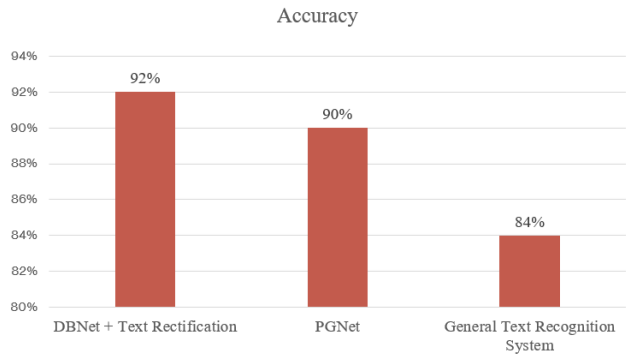


Figure 21: Scene Curved Text Image Recognition Accuracy Results

Experiments were conducted on a variety of scene curved text images from CTW1500 and a self-collected set, representative of various possible curvatures and backgrounds for scene texts in our daily lives. The DBNet text detection and text rectification algorithm achieved the highest recognition accuracy at 92%. It is 8% higher accuracy than a popular general text recognition system and a 2% higher accuracy than PGNet’s one-stage text recognition algorithm. The PGNet text recognition algorithm achieved an average recognition accuracy of 90%. Lastly, the popular general text recognition system achieved an accuracy around 84% (Figure 21).

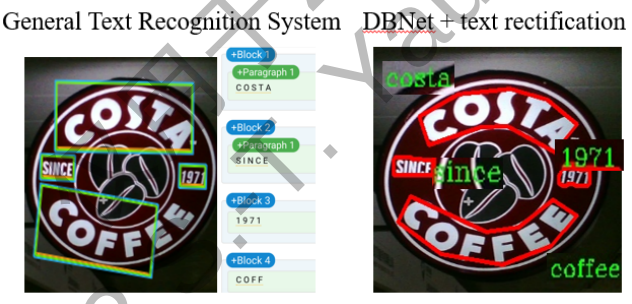


Figure 22: Visual Comparison of Text Recognition Systems

This visual comparison of the results clearly demonstrate how DBNet’s bounding boxes can surround each text instance more precisely compared to the rectangular bounding boxes of the general text recognition system (Figure 22).

4.3 Efficiency Experiment

The efficiency experiment was conducted with the same scene curved text images and text recognition systems. For the DBNet plus text rectification and the PGNet optimized by OpenVINO, the response times for each image and the average response time for all images were returned. For the general text recognition system, a timer was used to record the response time of each image. Then, the average response times were compared with each other.

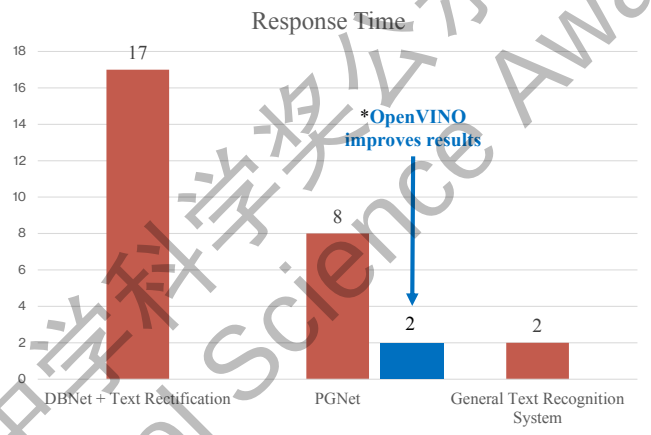


Figure 23: Scene Curved Text Image Recognition Response Time Results

DBNet has a response time of around 15-20 seconds per image. Prior to OpenVINO optimization, PGNet’s response time was at around 8 seconds per image. After OpenVINO is leveraged, PGNet’s speed increased to around 2 seconds per image, the same response time as the general text recognition system (Figure 23).

Therefore, PGNet has the best balance between accuracy and speed, especially after optimization with OpenVINO. Although PGNet has a network latency of approximately 1 second when it is deployed onto a server, it is still able to return the scene curved text recognition results within the acceptable range of under 5 seconds. Hence, it would be convenient for the visually impaired to use outside along with the mobile APP that I have developed. For recognition at home, I communicated with visually impaired people and learned that they would greatly appreciate any solution with a higher accuracy above 90%. So, although the DBNet algorithm is comparatively slower than PGNet and the popular general text recognition systems, it fulfills the needs of visually impaired people for their usage at home, such as recognizing a medicine bottle’s text label. This is crucial because they are unable to recognize the scene curved texts by themselves. Since the DBNet and PGNet algorithms are trained with scene curved text images, it is uniquely advantaged for solving this specific issue.

Not only are both the DBNet and PGNet algorithms more advantageous due to a higher scene curved text recognition accuracy than common modern solutions, but they also target the needs of visually impaired people with speech modules. Speech modules make my system more advantageous than current popular text recognition solutions for the visually impaired, which are based on

magnifying the text. They cannot address the difficulty of scene curved text recognition because the magnified texts are still curved, making it difficult to read. Furthermore, magnifying text does not work well in our daily lives, because recognizing restaurant names down the road with magnifying tools is neither convenient nor effective, especially without speech modules. Therefore, this system is more effective and accessible than current popular solutions for the visually impaired in addressing their difficulty of scene curved text recognition.

4.4 Trial with Visually Impaired People

After developing my prototype, I conducted trials with visually impaired people to receive some feedback from them. After testing out my product, they all found it very meaningful because it allows them to read scene text conveniently in their daily lives. They mentioned how many of the current products that have these functions are either inaccurate with scene curved text or are extremely expensive. In particular, Jack, a visually impaired high schooler with macular vision, volunteered to join me in the development of my product after testing out the prototype, because he was very satisfied and attracted to this project idea (Figure 24). As a result, we have been collecting scene text images together to train my model in the future, and optimize the dataset and text detection model for the needs of the visually impaired.

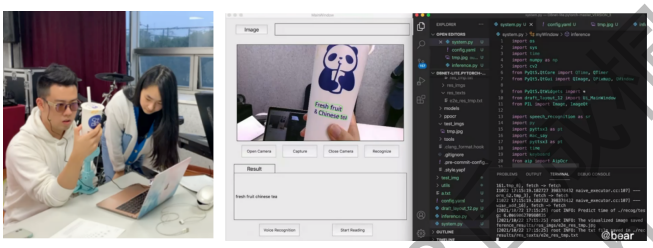


Figure 24: Trial with Visually Impaired High Schooler (Jack)

5 Conclusions

Firstly, scene curved text is prevalent in our daily lives because we often need to recognize restaurant signs, curved labels on packages, signs at places of public transportation, etc. The visual experiment provided convincing evidence that scene curved texts have a significant negative impact on visually impaired people's reading capabilities (see Appendix).

Secondly, my vision assistance system based on cloud-client architecture for the visually impaired includes a user-friendly APP on the mobile side and scene curved text recognition algorithms on the server end. For scene curved text recognition, my system provides two text detection and recognition algorithms with different balances between accuracy and speed. DBNet text detection with text rectification can recognize scene curved text at a high accuracy of 92% with a response time of 15-20 seconds, fulfilling visually impaired people's needs for recognition at home. My optimized PGNet fulfills their requirement for outdoor recognition, with a 90% scene curved text recognition accuracy and a response time of 2 seconds. Aside from scene curved text recognition, my system's APP also offers object, currency, and face recognition that call on Baidu's AI service. This system is unique from current popular text recognition solutions because it is a system that caters to the needs of visually impaired citizens with speech recognition and speech synthesis modules.

Lastly, I conducted trials with visually impaired people using my prototype. I received positive feedback from them and ophthalmologists as they mentioned that this is a more accurate, convenient, and affordable system.

6 Future Work

This paper has proposed a vision assistance system for the visually impaired, addressing their needs of scene text recognition in both outdoors and at home situations along with other categories of recognition. There are 3 key areas of future work: introducing vision language navigation, optimizing the dataset and text recognition model, and open source my system and source code to collaborate with others.

6.1 Vision Language Navigation

Since navigating and finding directions is also a major difficulty in the daily lives of visually impaired people, I hope to incorporate a Vision Language Navigation module for them to have accessible directions based on the immediate surroundings of visually impaired users. I also hope to include action recognition into the system so that they can be informed of the actions of others nearby.

6.2 Optimize Dataset and Model

Although the datasets ICDAR-2015 and CTW1500 contain bilingual scene and curved text, they are not specifically modified to suit the needs of the visually impaired. To enhance the user experience of my target audience, I am currently working with Jack, the visually impaired high schooler who volunteered to join my project, and we have been collecting scene text images that visually impaired citizens want to recognize in their daily lives. I hope to work with more visually impaired people to collect similar images. After training my model with this new dataset, the text recognition model will be more targeted to their needs. By interacting with visually impaired people, I also hope to further understand their needs.

6.3 Open Source for Collaboration

I have open sourced my system and source code on Github to allow other AI developers to access it. I am looking forward to collaborating with others who will build onto what I currently have to improve my current scene curved text recognition system. I also hope other AI developers can add in other types of recognition into my mobile APP, including traffic recognition and visual action recognition, to help the visually impaired. Here is the link to my project's source code on Github:

https://github.com/jasmine6524/Scene-Curved-Text-Recognition-System

7 Acknowledgment

Since 2020, I have been teaching visually impaired students English. Through communicating with them, I learned that scene text recognition is very difficult for them, especially when trying to

read labels on packages, restaurant signs down a street, and direction and warning signs. To help them reduce their inconvenience, I decided to create this assistance system with scene curved text recognition.

I would like to thank Professor Wen Guang Chen and Chang Song Liu (both are uncompensated) for offering valuable support and suggestions throughout the development of my project. They assisted me in the project direction, provided me with information on the newest available technology, guided me through my experiment, and coached me in refining this paper.

When I was developing my visual acuity chart, Dr. Xue Feng [38, 39] (uncompensated) from Eye and ENT Hospital of Fudan University informed me of the types of visual impairments, visual acuity charts, and continuously supported me. I am grateful for the assistance of Intel engineer Jing Zhang and Baidu AI engineers Xiaoting Yin and Yehua Yang (all are uncompensated). Finally, I am grateful for the cooperation of the visually impaired people (uncompensated) who participated in my visual acuity experiment and offered me precious feedback that significantly contributed to the final completion of my project.

References

[1] Ackland, Peter, et al. “World Blindness and Visual Impairment: Despite Many Successes, the Problem Is Growing.” Community Eye Health, International Centre for Eye Health, 2017, https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5820628/.

[2] Sterna, David J. “The Wide Spectrum Of Visual Impairment.” *David J. Sterna, OD & Associates, LLC*, 10 Oct. 2018, https://visionsource-davidsternaod.com/2018/10/10/the-wide-spectrum-of-visual-impairment/.

[3] “OrCam MyEye 2.0 - for People Who Are Blind or Visually Impaired.” *OrCam*, orcam.com/en/myeye2/.

[4] “Low-Vision Aids.” *American Academy of Ophthalmology*, 4 Mar. 2021, https://www.aao.org/eye-health/diseases/low-vision-aids

[5] “AngelEye.” *AngelEye Global*, 2020, www.angeleyeglobal.com/.

[6] Wang, Wenhai, et al. "Shape robust text detection with progressive scale expansion network." *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition.* 2019.

[7] Lyu, Pengyuan, et al. "Mask textspotter: An end-to-end trainable neural network for spotting text with arbitrary shapes." *Proceedings of the European Conference on Computer Vision (ECCV).* 2018.

[8] Girshick, Ross. "Fast r-cnn." *Proceedings of the IEEE international conference on computer vision.* 2015. Mask R-CNN:Mask r-cnn

[9] Ren, Shaoqing, et al. "Faster r-cnn: Towards real-time object detection with region proposal networks." *Advances in neural information processing systems* 28 (2015).

[10] He, Kaiming, et al. "Mask r-cnn." *Proceedings of the IEEE international conference on computer vision.* 2017.

[11] Xie, Enze, et al. "Scene text detection with supervised pyramid context network." *Proceedings of the AAAI conference on artificial intelligence.* Vol. 33. No. 01. 2019.

[12] Yuliang, Liu, et al. "Detecting curve text in the wild: New dataset and new solution." *arXiv preprint arXiv:1712.02170* (2017).

[13] Ch’ng, Chee Kheng, and Chee Seng Chan. "Total-text: A comprehensive dataset for scene text detection and recognition." *2017 14th IAPR international conference on document analysis and recognition (ICDAR).* Vol. 1. IEEE, 2017.

[14] “Overview - ICDAR2017 Competition on Multi-Lingual Scene Text Detection and Script Identification - Robust Reading Competition.” *Robust Reading Competition*, 2017, rrc.cvc.uab.es/?ch=8.

[15] Liao, Minghui, et al. "Real-time scene text detection with differentiable binarization." *Proceedings of the AAAI conference on artificial intelligence.* Vol. 34. No. 07. 2020.

[16] He, Kaiming, et al. "Deep residual learning for image recognition." *Proceedings of the IEEE conference on computer vision and pattern recognition.* 2016.

[17] Lin, Tsung-Yi, et al. "Feature pyramid networks for object detection." *Proceedings of the IEEE conference on computer vision and pattern recognition.* 2017.

[18] Dai, Jifeng, et al. "Deformable convolutional networks." *Proceedings of the IEEE international conference on computer vision.* 2017.

[19] Sun, Yipeng, et al. "Textnet: Irregular text reading from images with an end-to-end trainable network." *Asian Conference on Computer Vision.* Springer, Cham, 2018.

[20] Liu, Yuliang, et al. "Abcnet: Real-time scene text spotting with adaptive bezier-curve network." *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition.* 2020.

[21] Feng, Wei, et al. "Textdragon: An end-to-end framework for arbitrary shaped text spotting." *Proceedings of the IEEE/CVF international conference on computer vision.* 2019.

[22] Wang, Pengfei, et al. "PGNET: Real-time arbitrarily-shaped text spotting with point gathering network." *Proceedings of the AAAI Conference on Artificial Intelligence.* Vol. 35. No. 4. 2021.

[23] Xing, Linjie, et al. "Convolutional character networks." *Proceedings of the IEEE/CVF international conference on computer vision.* 2019.

[24] Wang, Zhongdao, et al. "Linkage based face clustering via graph convolution network." *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition.* 2019.

[25] Zhang, Shi-Xue, et al. "Deep relational reasoning graph network for arbitrary shape text detection." *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition.* 2020.

[26] Graves, Alex, et al. "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks." *Proceedings of the 23rd international conference on Machine learning.* 2006.

[27] OpenVINO. “Tutorials.” OpenVINO, https://docs.openvino.ai/latest/tutorials.html.

[28] Karatzas, Dimosthenis, et al. "ICDAR 2015 competition on robust reading." *2015 13th international conference on document analysis and recognition (ICDAR).* IEEE, 2015.

[29] Zhang, Zheng, et al. "Multi-oriented text detection with fully convolutional networks." *Proceedings of the IEEE conference on computer vision and pattern recognition.* 2016.

[30] SynthText:Gupta, Ankush, Andrea Vedaldi, and Andrew Zisserman. "Synthetic data for text localisation in natural images." *Proceedings of the IEEE conference on computer vision and pattern recognition.* 2016.

[31] Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E. Hinton. "Imagenet classification with deep convolutional neural networks." *Communications of the ACM* 60.6 (2017): 84-90.

[32] Liao, Minghui, et al. "Rotation-sensitive regression for oriented scene text detection." *Proceedings of the IEEE conference on computer vision and pattern recognition.* 2018.

[33] Vatti, Bala R. "A generic solution to polygon clipping." *Communications of the ACM* 35.7 (1992): 56-63.

[34] Hu, Wenyang, et al. "Gtc: Guided training of ctc towards efficient and accurate scene text recognition." *Proceedings of the AAAI Conference on Artificial Intelligence.* Vol. 34. No. 07. 2020.

[35] Girshick, Ross, et al. "Rich feature hierarchies for accurate object detection and semantic segmentation." *Proceedings of the IEEE conference on computer vision and pattern recognition.* 2014.

[36] Pelli, Dennis, et al. "Crowding and eccentricity determine reading rate." *Journal of Vision*, Oct. 2007, jov.arvojournals.org/article.aspx?articleid=2122073.

[37] Wang, Chen-xiao, and Lü, Fan. "Selection of Chinese characters for visual acuity charts using psychophysical methods." *Pubmed*, National Library of Medicine, July 2011, pubmed.ncbi.nlm.nih.gov/22041487/.

[38] Zhang, Jun-Yun, et al. "Legibility of Chinese characters in peripheral vision and the top-down influences on crowding." *Pubmed*, National Library of Medicine, 7 Nov. 2008, pubmed.ncbi.nlm.nih.gov/18929592/.

[39] Zhang, Jun-Yun, et al. "Legibility Variations of Chinese Characters and Implications for Visual Acuity Measurement in Chinese Reading Population." *Investigative Ophthalmology & Visual Science*, May 2007, iovs.arvojournals.org/article.aspx?articleid=2164478.

[40] "Psychophysics of reading. XV: Font effects in normal and low vision." *Pubmed*, National Library of Medicine, 1996, pubmed.ncbi.nlm.nih.gov/8675391/.

[41] "Visual requirement for Chinese reading with normal vision." *Wiley Online Library*, 21 Feb. 2019, onlinelibrary.wiley.com/doi/10.1002/brb3.1216.

[42] "The key parameters of design research and analysis of the Chinese reading visual acuity chart." *Pubmed*, National Library of Medicine, June 2013, pubmed.ncbi.nlm.nih.gov/24119968/.

[43] "Design principles for reading-acuity charts and their implementation in the MNREAD charts." *MNREAD Acuity Charts*, University of Minnesota, Dec. 2020, mnread.umn.edu/design.

Appendix

A Visual Acuity Experiment

To determine how much text curvature decreases visually impaired people's reading ability, I conducted an experiment with visually impaired people who provided voluntary consent.

A.1 Background Research

Current research focuses on two main types of factors that decrease reading ability. The first is factors related to the characters' environment. This includes the luminosity, contrast, and color of characters. The second category is factors related to the characters themselves, including the font type, font size, and complexity of characters [39–41]. Minimal research has been done in the area of the spatial layout of characters.

Previous works have investigated in spatial factors including complexity, eccentricity, and text curvature [36–38]. Pelli et al. focuses on people's peripheral vision, concluding that a larger vertical distance between the point of visual fixation and the target text recognition region, known as eccentricity, decreases reading speed [36]. This connects to our study because curved text contains letters that are vertically distant from each other and slanted at an angle. Hence, an increase in vertical distance, or curvature, should lead to a decrease in reading speed and accuracy. Wang and Lü have developed a Chinese Character Visual Acuity Chart. This study and other related works have analyzed how font size, font type, and within-character spatial complexity can affect our text recognition [37, 42, 43]. Since the characters in this visual acuity chart have similar influence on people's reading ability, they have been incorporated into our experiment. Our experiment investigates how the factor of text curvature impacts our reading ability because it is commonly seen in our daily lives.

A.2 Curved Text Visual Acuity Chart:

I created a curved text visual acuity chart, modeled upon the *Standard for Logarithmic Visual Acuity Chart for low-vision people, GB11533-2011*. This chart has the same layout as the internationally employed LogMAR reading chart. The numeric representation of a subject's vision is measured through decimal values, ranging from 0.05 to 1.2 in this chart. The subject stands 40cm away from the reading chart when testing their vision. The font sizes of characters for this study's reading acuity chart were selected based on the *Chinese Reading Visual Acuity Chart* that corresponds to the different font sizes with values on the *Standard for Logarithmic Visual Acuity Chart, GB 11533-2011*. The sizes are 33, 26, 21, 16.5, 13, and 10.5, each matching a visual ability measure (Figure 25).

A.3 Results

After conducting this experiment with 10 visually impaired volunteers who voluntarily consented to the experiment, the results

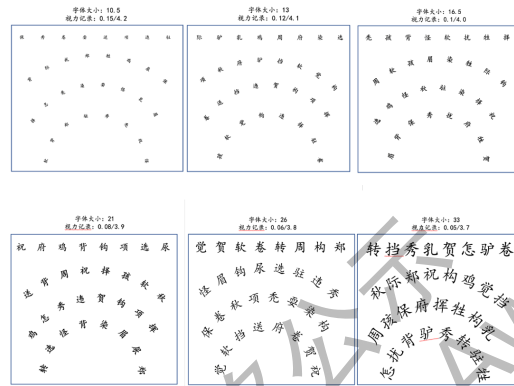


Figure 25: Novel Curved Chinese Character Visual Acuity Chart

sorted into three groups show an average decrease in accuracy by 44% and an average decrease in speed by 66% as curvature increases. This demonstrates that an increase in text curvature significantly decreases people's reading ability (Figure 26).

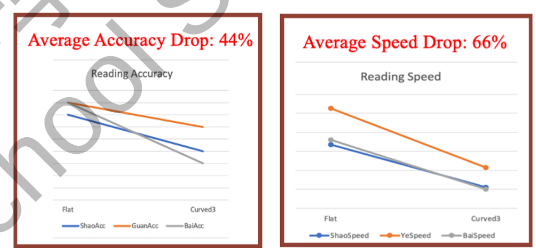


Figure 26: Vision Experiment Results

B Mobile APP Recognition Categories

Overall, this paper provides an assistance system for the visually impaired. The tiny camera can be attached to any ordinary pair of glasses, making this system easy to use when travelling around outside.

Not only is scene curved text recognition included, but other categories of recognition are also incorporated to assist visually impaired people in their daily lives. These include object, currency, and face recognition. They are called upon through HTTP to access Baidu's AI service. For example, object recognition can identify the type of object that is seen, such as a car or a computer, while currency recognition can recognize the type and value of any given currency. Below are 2 examples. When visually impaired people can recognize the objects in their surroundings they will encounter less difficulties with mobility. When shopping, being able to recognize the products and paper money will be especially important. Visually impaired people will also be more comfortable with integrating into our society and communicating with others when they are able to accurately recognize faces and greet the people around them. Hence, this system can not only recognize scene curved text with a higher accuracy than existing smart glasses, but it can also perform the other functions that these smart glasses have.

The recognized results are read aloud with speech synthesis for visually impaired users to hear. Compared to existing smart glasses, the combination of recognition functions are achieved in this system at a much more affordable price by taking advantage of cloud computing and phone computing power. Overall, this affordable and convenient system will allow visually impaired people to become more independent in their daily lives, improving their overall living standards.