DISPLAY OF A SYSTEM OF REPRODUCTION OF PRINTED TEXT FOR PEOPLE WITH VISUAL IMPAIRMENT

Slađan Kantar[1], Igor Medenica[1], Jelena Vasiljević[2,3], Miloš Jovanović[2,3], Đorđe Babić[2] , Dhinaharan Nagamalai[4]

[1]Faculty of Computer Science, University Union, Serbia
[2]Faculty of Computer Science, University of Belgrade, Serbia
[3] Institute Mihajlo Pupin, Serbia
[4]Wireilla, Australia

ABSTRACT

Recent technologies are being used to facilitate and speed up the way in which printed materials are reproduced for people with visual impairment. In this paper, one of the audio playback methods will be described. With the help of the Google Assistant SDK and the OCR Space Libraries, an automated system is created that forwards the text recognized via webcam, through the local web server to the device that plays the text in segments. The goal of this system is to facilitate and improve the way in which people with this type of disability read and learn.

KEYWORDS

BeagleBone Black; OCR; Google Assistant SDK; NodeJS

1. INTRODUCTION

The rapid development of information and communication technologies has contributed to the introduction of numerous innovations that make it easier for people with disabilities to minimize their shortcomings. People with visual impairment are not able to process texts, so they need tools that convert such a recording into one that they can process. Audio recording is the most popular. Text to speech (TTS) is the oldest speech technology whose beginnings date back to the 18th century, when the first "talking machines" appeared. The task of TTS is to generate, based on input information in a textual form, a speech signal that is understandable to a human being. In the meantime, there has been a dramatic development in that area, largely thanks to the development of computer technology in the last few decades. This is a speech technology where language dependence is largely expressed, so the speech synthesizer for each new language needs to be realized practically from the beginning, and possible adaptations are only possible for extremely similar languages.

Speech technologies have achieved a level of development that allows many useful applications. One of the applications that no longer raises the question of whether it has matured, is the one that is used to help people with disabilities. However, there is only a few such small speaking areas for which speech technologies have been developed so much that they can concretely, practically and permanently help in realizing the basic human rights of these people.

For voice technologies to be widely used to help people with disabilities, it is essential that they be made available in their native language. Through this system, we are trying to solve the problem of recognizing text through the webcam and its communication with the device that plays the text in audio format. As described in, important problems in the system are downloading text and connecting to a playback device. [1-4]

During the research on the BeagleBone Black board, a camera that recognizes the text was attached and a server was created that sends the recognized content as a response, from the server to the device which plays the content.

The purpose of this system is to facilitate and speed up the reading of written literature for people with visual impairment. This system uses the Google SDK provided by Google to communicate with the server. The paper is organized in the following form: Section 2 describes the hardware components used in this paper. Section 3 represents the course of the complete solution; Section 4 describes an OCR component as one of the major components of this work; while Section 5 gives a brief conclusion as well as ideas for further development.

2. Components Used In The Work

For implementation of the presented system, due to its advantages, a google home device (Figure 1) was used, with BeagleBone black development panel (Figure 2), as well as Logitech C170 webcam (Figure 3).

2.1. Google Home

The device is a Google product, and is a part of Google Home products which are applied in smart homes. It was named Google Home Mini due to its dimensions. This is a smart speaker that uses Google Assistant.

Google Home Mini is not specifically designed to play music and to communicate with devices that are used daily in the household. With the Google Assistant SDK, it is possible to control the execution of various tasks. One of the functions used in this research is the reproduction of forwarded textual content. The device is shown in the picture below (Figure 1). [5-6]

Figure 1. Google Home Mini

2.2. BEAGLEBONE BLACK

BeagleBone Black is a development platform, with a storage space of 4GB. It has a micro HDMI port on board, 512MB DDR3L DRAM memory, AM3358 processor on 1GHz and JTAG optional interface. It has a large processing power for real-time analysis provided by the TI Sitara ™ AM3358 ARM® Cortex ™ -A8 processor. It can also be supplemented with "cape" plug-ins that increase its functionality. It has a Linux operating system installed on it. Display of the development panel on the image below (Fig. 2).

Figure 2. BeagleBone black

2.3. C. LOGITECH C170

We used Logitech C170 webcam because of compatibility with Linux OS. The resolution of the images taken is 1024x768. The device is shown in the picture below (Figure 3).

Figure 3. Web camera Logitech C170

3. DESCRIPTION OF THE SOLUTION

The complete solution consists of several independent components, integrated into one whole. Components can be grouped into two abstract entities (Figure 4):

- frontend - composed of Google Home device with its integrated functionalities
- backend - composed of BeagleBone device and camera

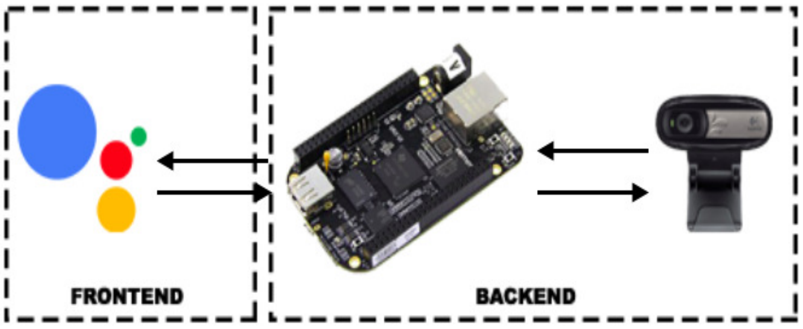


Figure 4. System components. Arrows represent the direction of communication between individual devices.

The Google Assistant transfers the command to the BeagleBone device that manages the camera. After retrieving the image from the camera, it extracts the text, and as such returns it to the Google Home device, for playback.

3.1. Frontend Part

The Frontend part of this solution is the Google Assistant with all the features that accompany it. The Google Assistant can be found in many forms, such as an application on smartphones (Android, iOS), a Google Home device, etc.

Google Assistant represents a connection between the user and the system itself. With Google Home Assistant, user can issue commands to the system, and the system can give an audio response.

Google Assistant contains an interface for creating custom commands and defining actions that will be executed after identifying the specified command [7]. Each command belongs to an agent, thus opening the possibility of creating different agents for controlling different parts of smart houses.

Implementation of the Frontend component is done in several steps:

- creating a new Agent for Google Assistant devices
- creating a new command within a pre-created agent
- teaching the phrases after which the command will be executed
- defining the way of communication between the Google Home device and the backend part

3.2. Backend Part

Backend part of the system consists of two hardware components (BeagleBone black, camera), and several software components. (NodeJS server, OCR ...).

The complete module runs on the BeagleBone Black board on which the Linux operating system is installed. After upgrading the operating system, the BeagleBone board is running Linux Debian 9.5. The implementation of the Backend logic was carried out in the following steps:

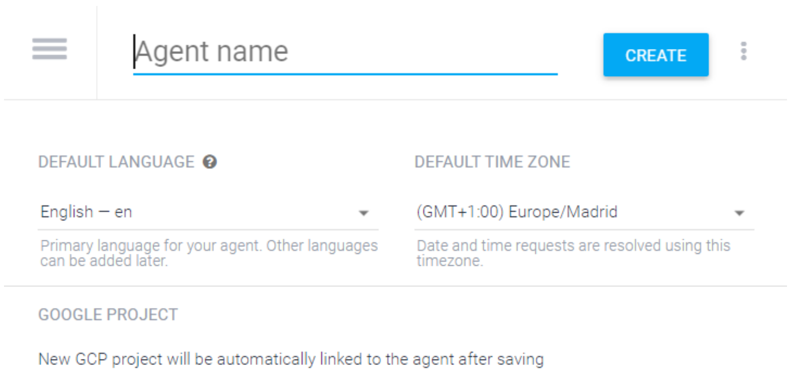
- camera configuration and installation of additional software solutions for working with the camera

- implementation of the NodeJS server for processing requests received from Google Assistant
- configuring a public address with an SSL protocol and establishing communication with Google Assistant
- implementation of the OCR component for extracting text from the image

4. DESCRIPTION OF THE FRONTEND PART

4.1. CREATING A NEW AGENT FOR GOOGLE ASSISTANT

Creating an agent is done by simply filling in the necessary information such as name, language, time zone, etc. (Figure 5)



DEFAULT LANGUAGE

English – en

Primary language for your agent. Other languages can be added later.

DEFAULT TIME ZONE

(GMT+1:00) Europe/Madrid

Date and time requests are resolved using this timezone.

GOOGLE PROJECT

New GCP project will be automatically linked to the agent after saving

Figure 5. Appearance of the Google Assistant development environment/section for creating new agents.

Each successfully created agent comes with three pre-trained commands:

- default welcome intent - the command that starts the communication with the defined agent (the command has the form: "Hey Google, talk to my “Agent Name”)
- default fallback intent - is triggered as a response to all commands incomprehensible to the system (contains messages such as: “Sorry, I did not get that.”; “Can you rephrase”; “I missed that, say that again” ....)
- default exit intent - a command that terminates communication with an agent (command: “Cancel”)

After a successfully created agent, all Google Assistant devices can use previously defined commands, and the user has the ability to create / train new commands.

4.2. CREATING A NEW COMMAND WITHIN AN AGENT

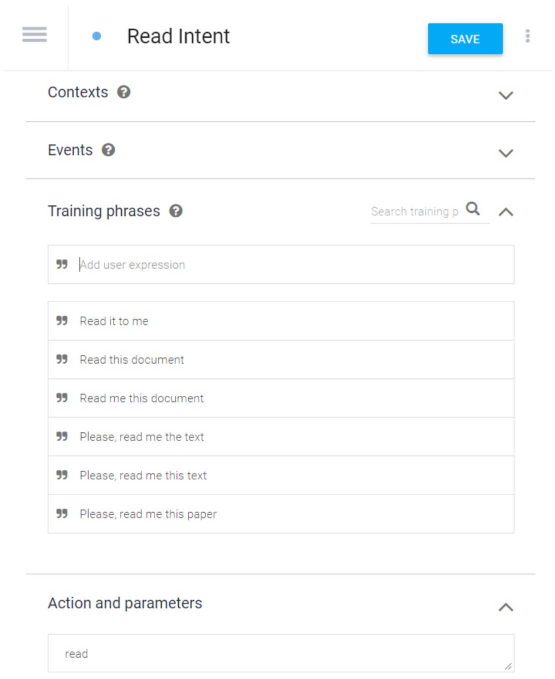
Within the agent, in the Intents section, all the commands that the current agent can handle are listed, as well as the clearly defined option for creating a new action (New Intent). Each new intent requires information such as: Phrases for triggering a command, actions that are taken in response to the triggered command. (Figure 6)

4.3. Creating A New Command Within An Agent

Under the Training phrases section, we enter the phrases that Google Assistant uses to train an integrated voice recognition algorithm.

For this command we used phrases such as (Figure 6):

- Please, read this document
- Read this document
- Read this
- Read



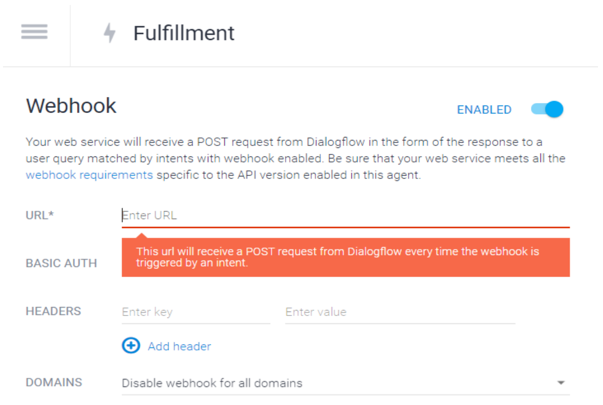
Read it to me
Read this document
Read me this document
Please, read me the text
Please, read me this text
Please, read me this paper

read

Figure 6. Appearance of the Google Assistant Development Environment / section for creating new commands. In the list we see the phrases used to train the intent used in this paper.

4.4. Defining The Way Of Communication Between Google Home Device And The Backend Part

Google home agent can create a webhook after each recognized command. (Figure 7)



Fulfillment

Webhook ENABLED

Your web service will receive a POST request from Dialogflow in the form of the response to a user query matched by intents with webhook enabled. Be sure that your web service meets all the webhook requirements specific to the API version enabled in this agent.

URL* Enter URL

This url will receive a POST request from Dialogflow every time the webhook is triggered by an intent.

BASIC AUTH

HEADERS Enter key Enter value

Add header

DOMAINS Disable webhook for all domains

Figure 7. Appearance of the Google Assistant Development Environment / section for defining a web hook for an agent

After successfully defining the URL parameter, after each command recognized by the Google Assistant, a post request will be sent to the defined address. Post request as one of the arguments sends the parameter defined in the section Actions and parameters (Figure 5). This parameter allows the backend page to distinguish between actions.

Previous steps have enabled our system to receive a voice command by the user (trained phrases), send information to our backend part of the system (webhook), and wait for a string response from the system, which will be automatically played back.

5. Description Of The Backend Part

5.1. Camera Configuration And Installation Of Additional Software Solutions For Working With The Camera

Recent versions of operating systems do not require additional software components that would allow the operation of hardware components such as the camera. By upgrading the version of the operating system, the camera is automatically recognized as an additional hardware device. The only functionality needed to implement this work is image capturing.

There are several software solutions that allow interaction with the camera via the command line. For the purposes of this paper we used the fswebcam tool.

fswebcam is one of the most popular tools for using the command line to control the camera. The tool contains many options, out of which we only used a few. A complete command used within the module, as well as explanations of the options, is shown in Figure 8, the command captures the current output of the camera and places the image in the working directory.

```
fswebcam -r 640x480 -S 10 -F 5 --no-banner imageFile.jpg'
```

Figure 8. Command for taking pictures with the camera. r defines the resolution of the image; s defines the skipping of the first 10 frames in order for the camera to calibrate, or to get the best picture quality. f the number of frames from which the image will be calculated. no-banner argument that neutralizes the date added over the image (the date would be recognized by the OCR module, and this effect is not desired) imageFile arbitrary image name.

5.2. Implementation of the NodeJS server for processing requests received from Google

As one of the basic tasks of NodeJS framework is the creation of a web server, this task was not overly complicated. Also, an additional facilitator of communication between server and Google Assistant is the opensource action-on-google library designed by the google engineering team. The library is designed specifically for the purpose of creating arbitrary components that communicate with Google Assistant.

Upon reception of the request, it is necessary for the server to synchronously execute the following steps:

- default capturing an image with a camera using fsweb tool
- OCR processes the received image, and extracts the text from it
- creates a response in the required format, with string received from the OCR engine.

5.3. Configuring a public address with an SSL protocol and establishing communication with Google Assistant

As the Google Assistant framework allows communication via webhook only with a public address using the SSL certificate, the server implemented in this work had to meet these requirements. Therefore, the webhook link had to start with the prefix https.
To testing, a simple ngrok tool was used. The main task of this tool is to quickly and automatically create a random public URL address through which it is possible to communicate with the local server. An explanation of how the tool works can be found on the official Internet address of the tool.

After this step, an automatically received address is assigned to an application created within the Google Assistant IDE, and communication between the server and the Google Assistant device is established.

6. Optical Character Recognition

Tesseract is a free software for recognizing text in the image, available under the Apache2 license. It represents one of the most popular and most used OCR Engines, which was a good enough argument for the purpose of this work.[11]

After installing a free tool, and testing it with images downloaded from the camera, the results were extremely defeating. Namely, the software recognized less than 30% of the words and as such it was not of great help. After a short survey, it was noticed that the tool contains certain restrictions that had to be respected. This is precisely the fact that caused the additional pre-processing of the input image.

To achieve the most accurate results, the pre-processing of the image is implemented in several steps:

- removing the noise in the image
- neutralizing image distortion
- extracting a surface which contains text
- measuring success

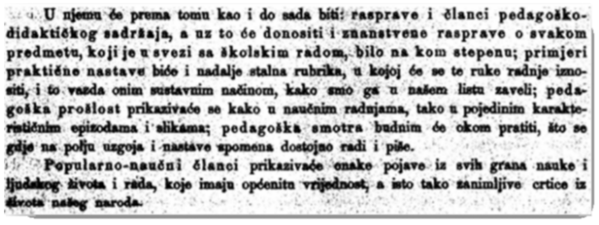
6.1. REMOVING NOISE

One of the biggest problems encountered in digital image processing is noise. Noise occurs in most processes that involve the transmission or modification of the media. For example, converting information on paper to digital form can lead to noise appearance.

The methods of noise reduction in general try to reduce its influence to better apply various algorithms. This can be mathematically shaped and appropriate noise reduction methods can be implemented.

In the work itself, some classic filtering methods have been applied which involve the use of linear (uniform, triangular, Gaussian, Vinner ...) and non-linear filters (median, Kuwahara). Since the images of documents distorted by noise are usually subjected to both filtering and binarization, in order to separate objects from the background, the paper presents an algorithm that does exactly that.

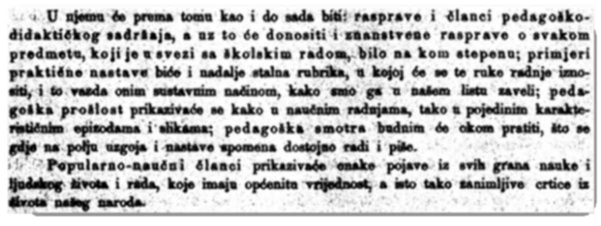
The noise removal is done by a uniform, median and Gaussian filter. The original image was shown in Figure 9 and result after filter in Figure 10.[10]



U njemu će prema tomu kao i do sada biti: rasprave i članci pedagoško-didaktičkog sadržaja, a uz to će donositi i znanstvene rasprave o svakom predmetu, koji je u svezi sa školskim radom, bilo na kom stepenu; primjeri praktične nastave biće i nadalje stalna rubrika, u kojoj će se te ruke radnje iznositi, i to vazda onim sustavnim načinom, kako smo ga u našem listu zaveli; pedagoška prošlost prikazivaće se kako u naučnim radnjama, tako u pojedinim karakterističnim epizodama i slikama; pedagoška smotra budnim će okom pratiti, što se gdje na polju uzgoja i nastave spomena dostojno radi i piše.

Popularno-naučni članci prikazivaće znake pojave iz svih grana nauke i ljudskog života i rada, koje imaju općenitu vrijednost, a isto tako zanimljive crtice iz života našeg naroda.

Fig. 9. Image with text before noise removal



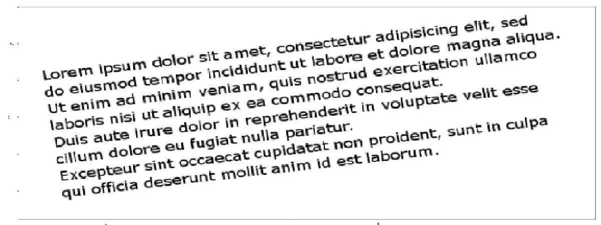
U njemu će prema tomu kao i do sada biti: rasprave i članci pedagoško-didaktičkog sadržaja, a uz to će donositi i znanstvene rasprave o svakom predmetu, koji je u svezi sa školskim radom, bilo na kom stepenu; primjeri praktične nastave biće i nadalje stalna rubrika, u kojoj će se te ruke radnje iznositi, i to vazda onim sustavnim načinom, kako smo ga u našem listu zaveli; pedagoška prošlost prikazivaće se kako u naučnim radnjama, tako u pojedinim karakterističnim epizodama i slikama; pedagoška smotra budnim će okom pratiti, što se gdje na polju uzgoja i nastave spomena dostojno radi i piše.

Popularno-naučni članci prikazivaće znake pojave iz svih grana nauke i ljudskog života i rada, koje imaju općenitu vrijednost, a isto tako zanimljive crtice iz života našeg naroda.

Fig. 10. Image with text after noise removal

6.2. REMOVING IMAGE DISTORTION

As the product of this work has no restrictions on placing the text in the camera's viewport, the text in the image captured by the camera may contain a certain distortion. The aim of this section is to remove the angle of rotation and enable the extraction of the text from the image. An example of a text containing a specific angle of rotation is shown in Figure 11.



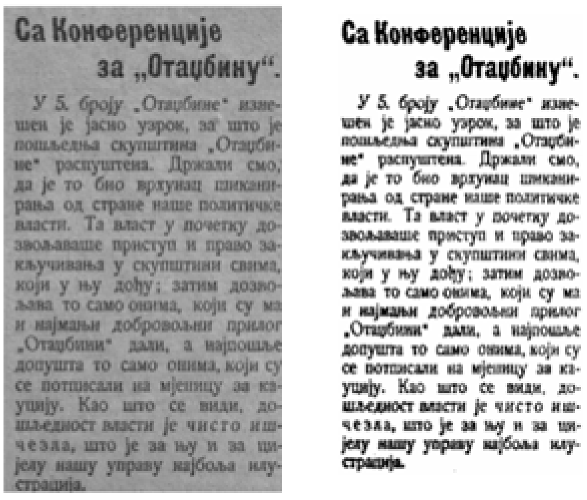
Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed
do eiusmod tempor incididunt ut labore et dolore magna aliqua.
Ut enim ad minim veniam, quis nostrud exercitation ullamco
laboris nisi ut aliquip ex ea commodo consequat.
Duis aute irure dolor in reprehenderit in voluptate velit esse
cillum dolore eu fugiat nulla pariatur.
Excepteur sint occaecat cupidatat non proident, sunt in culpa
qui officia deserunt mollit anim id est laborum.

Fig. 11. Image with rotated text

The first step in eliminating rotation is the image binarization. Although we predominantly see black and white pixels in the picture, we can spot gray pixels in the immediate vicinity of the text. The binarization is done by defining the values above which the pixels become black, or below which they receive a white color. [12] Binarization of image was shown in Figure 12.

After image binarization, we find a minimal surface that includes black pixels, from which we can easily get the angle of rotation.

Image binarization and finding the minimum surface of black pixels represent commonly known problems, and the OpenCV library contains functions that greatly facilitate calculations. These benefits precisely represented the argument for selecting the programming language in the creation.


Са Конференције
за „Отаџбину“.

У 5. броју „Отаџбине“ изне-
шен је јасно узрок, за што је
пошљедња скупштина „Отаџби-
не“ распуштена. Држали смо,
да је то био врхунац шикани-
рања од стране наше политичке
власти. Та власт у почетку до-
звољаваше приступ и право за-
кључивања у скупштини свима,
који у њу дођу; затим дозво-
љава то само онима, који су ма
и најмањи добровољни прилог
„Отаџбини“ дали, а најпошље
допушта то само онима, који су
се потписали на мјеницу за ка-
уцију. Као што се види, до-
шљедност власти је чисто иш-
чезла, што је за њу и за ци-
јелу нашу управу најбоља илу-
страција.

Са Конференције
за „Отаџбину“.

У 5. броју „Отаџбине“ изне-
шен је јасно узрок, за што је
пошљедња скупштина „Отаџби-
не“ распуштена. Држали смо,
да је то био врхунац шикани-
рања од стране наше политичке
власти. Та власт у почетку до-
звољаваше приступ и право за-
кључивања у скупштини свима,
који у њу дођу; затим дозво-
љава то само онима, који су ма
и најмањи добровољни прилог
„Отаџбини“ дали, а најпошље
допушта то само онима, који су
се потписали на мјеницу за ка-
уцију. Као што се види, до-
шљедност власти је чисто иш-
чезла, што је за њу и за ци-
јелу нашу управу најбоља илу-
страција.

Fig. 12. Image example before and after binarization.

The result after removing the angle of rotation is shown in Figure 13.

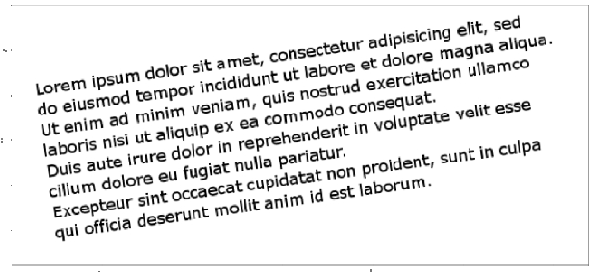

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed
do eiusmod tempor incididunt ut labore et dolore magna aliqua.
Ut enim ad minim veniam, quis nostrud exercitation ullamco
laboris nisi ut aliquip ex ea commodo consequat.
Duis aute irure dolor in reprehenderit in voluptate velit esse
cillum dolore eu fugiat nulla pariatur.
Excepteur sint occaecat cupidatat non proident, sunt in culpa
qui officia deserunt mollit anim id est laborum.

Fig. 13. Image after removing distortion

6.3. Extracting The Text Surface

The problem of removing the excessive part of the photo, which does not contain text, certainly greatly accelerates the later processing of the image and increases the accuracy. A binarized image, and a minimum surface containing the text detected in the previous chapter, is used to successfully extract the text. An example after removing the excess surface is shown in Figure 14.

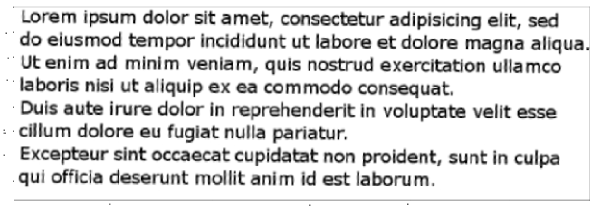

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed
do eiusmod tempor incididunt ut labore et dolore magna aliqua.
Ut enim ad minim veniam, quis nostrud exercitation ullamco
laboris nisi ut aliquip ex ea commodo consequat.
Duis aute irure dolor in reprehenderit in voluptate velit esse
cillum dolore eu fugiat nulla pariatur.
Excepteur sint occaecat cupidatat non proident, sunt in culpa
qui officia deserunt mollit anim id est laborum.

Fig. 14. Image after removing distortion

6.4. MEASURING SUCCESS

After starting the tool that extracts text from pre-processed image, the results are dramatically improved. Namely, on the dataset of 300 documents with labeled text, the word recognition efficiency is 87%.

7. CONCLUSION

In this paper, example of a system for voice reproduction of printed text is presented. The examples show custom device interaction with Google Assistant devices.

Based on research, for better results, it is necessary to train on several samples. Expanding the language base would make it easier for people with visual impairment to hear the audio signal being played in a language that is closer to their knowledge and needs. An improved system would support users in a better understanding of textual content and in enhancing the learning process.

ACKNOWLEDGEMENTS

This work was supported by the Ministry of Education and Science, Republic of Serbia, Science and Technological Development grant TR32037.

REFERENCES

[1] Interactive home automation system with Google Assistant, International Journal of Pure and Applied Mathematics, Volume 119, No.12, 2018

[2] P. Milhorat, S. Schogl, G. Chollet et all “Building the Next Generation of Personal Digital Assistants” 1st International Conference on Advanced Technologies

[3] V. Kepuska and G. Bohouta, “Next Generation of Virtual Personal Assistants (Microsoft Cortana, Apple Siri, Amazon Alexa and Google Home)”, 2018 IEEE 8th Annual Computing and Communication Workshop and Conference 8-10 Jan. 2018 Las Vegas, USA, pp.99–103.

[4] P. J. Young, J. H. Jin, S. Woo and D. H. Lee, “Bad Voice: Soundless Voice-control Replay Attack on Modern Smartphones”, 2016 Eigth International Conference on Ubiquitous and Future Networks (ICUFN), Vienna, Austria, pp. 882–887.

[5] Google official website of artificial intelligence projects, https://aiyprojects.withgoogle.com/voice/, accessed March 7, 2019

[6] Google cloud official site, https://cloud.google.com/speech-to-text/docs/languages accessed March 7, 2019.

[8] F.Weng, P. Angkititrakul, E. Shirberg et all, Conversational In-Vehicle Dialog Systems: The past, present, and future, IEEE Signal Processing Magazine, vol. 33, issue 6, pp. 49-60, 2016

[9] G. Nagy, "At the frontiers of OCR", Proc. IEEE 80(7), IEEE, U Smith. "An overview of the Tesseract OCR Engine." Proc 9th SA, Jul 1992, pp 1093-1100

[10] Archana A. Shide D. 2012.Text Pre-processing and Text Segmentation for OCR. International Journal of Computer Science Engineering and Technology, pp. 810-812.

[11] Smith R. 2007. An Overview of the Tesseract OCR Engine. In proceedings of Document analysis and International Journal of Computer Applications (0975 – 8887) Volume 55– No.10, October 2012 56 Recognition. ICDAR 2007. IEEE Ninth International Conference.

[12] Jian Liang; DeMenthon, D.; Doermann, D. Geometric Rectification of Camera-Captured Document Images. April 2008. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol.30, no.4, pp.591-605.

[13] R.Int. Conf. on Document Analysis and Recognition, IEEE, Curitiba, Brazil, Sep 2007, pp629-633.

[14] The Tesseract open source OCR engine, http://code.google.com/p/tesseract-oc

[15] Google Home document reader, https://www.youtube.com/watch?v=NqfgQGBO9Lg