electronics

MDPI

Article

Smart Glass System Using Deep Learning for the Blind and Visually Impaired

Mukhriddin Mukhiddinov and Jinsoo Cho *

Department of Computer Engineering, Gachon University, Sujeong-gu, Seongnam-si 13120, Korea; mukhiddinov18@gachon.ac.kr
* Correspondence: jscho@gachon.ac.kr

Abstract: Individuals suffering from visual impairments and blindness encounter difficulties in moving independently and overcoming various problems in their routine lives. As a solution, artificial intelligence and computer vision approaches facilitate blind and visually impaired (BVI) people in fulfilling their primary activities without much dependency on other people. Smart glasses are a potential assistive technology for BVI people to aid in individual travel and provide social comfort and safety. However, practically, the BVI are unable move alone, particularly in dark scenes and at night. In this study we propose a smart glass system for BVI people, employing computer vision techniques and deep learning models, audio feedback, and tactile graphics to facilitate independent movement in a night-time environment. The system is divided into four models: a low-light image enhancement model, an object recognition and audio feedback model, a salient object detection model, and a text-to-speech and tactile graphics generation model. Thus, this system was developed to assist in the following manner: (1) enhancing the contrast of images under low-light conditions employing a two-branch exposure-fusion network; (2) guiding users with audio feedback using a transformer encoder–decoder object detection model that can recognize 133 categories of sound, such as people, animals, cars, etc., and (3) accessing visual information using salient object extraction, text recognition, and refreshable tactile display. We evaluated the performance of the system and achieved competitive performance on the challenging Low-Light and ExDark datasets.

Keywords: smart glasses; artificial intelligence; blind and visually impaired; deep learning; low-light images; assistive technologies; object detection; refreshable tactile display

Citation: Mukhiddinov, M.; Cho, J. Smart Glass System Using Deep Learning for the Blind and Visually Impaired. *Electronics* **2021**, *10*, 2756. https://doi.org/10.3390/electronics10222756

Academic Editor: Amir Mosavi

Received: 27 September 2021
Accepted: 8 November 2021
Published: 11 November 2021

1. Introduction

In the modern era of information and communication technology, the lifestyle and independent movement of blind and visually impaired people is among the most significant issues in society that need to be addressed. Governments and various specialized organizations have enacted many laws and standards to support people with visual disabilities and have organized essential infrastructure for them. According to the World Health Organization, at least 2.2 billion people worldwide suffer from vision impairment or blindness, of whom at least 1 billion have a vision impairment that could have been prevented or is yet to be addressed in 2020 [1]. Vision impairment or blindness may be caused by several reasons, such as, cataract (94 million), unaddressed refractive error (88.4 million), glaucoma (7.7 million), corneal opacities (4.2 million), diabetic retinopathy (3.9 million), trachoma (2 million), and others [1]. The primary problems that blind and visually impaired (BVI) people encounter in their routine lives involve action and environmental awareness. Several solutions exist to such problems, employing navigation and object recognition methods. However, the most effective navigation methods, such as a cane, trained guide dogs, and smartphone applications suffer from certain drawbacks; for example, a cane is ineffectual over long distances, crowded places, and cannot provide

information regarding dangerous objects or car traffic when crossing the street, whereas training of guide dogs is cumbersome and expensive, and dogs require special attention when caring for them. Further, although smartphone applications such as voice assistance and navigation maps for BVI people are evolving rapidly, proper and complete use is still low.

Recent advancements in embedded systems and artificial intelligence have had a significant impact on the field of wearable assistive technologies for the visually impaired, and consequently several devices have been developed and placed on the market. Assistive systems exist to aid BVI people with navigation and daily activities, such as distinguishing banknotes [2,3], crossing a road [4–6], video media accessibility [7,8], image sonification for navigation [9,10], recognizing people [11–13], recognizing private visual information [14], selecting clothing [15], and navigating both outdoors and indoors [16–18]. Daescu et al. [13] proposed a face recognition system with smart glasses and used a server-based deep learning face recognition model. In this system, they used client–server architecture to reduce power consumption and computational time. Joshi et al. [17] introduced an assistive device to recognize different objects based on a deep learning model, and a distance-measuring sensor was combined to make the device more complete by identifying barriers while traveling from one place to another.

However, owing to the low-light environment and the lack of light at night, problems such as recognizing people and objects as well as providing accurate information to the BVI people are prevalent. Moreover, the visually impaired face serious challenges, particularly when walking in public places, where simple actions such as avoiding obstacles, crossing the street, and using public transportation can pose significant risks and challenges. Such challenges endanger the safe and confident independent action of BVIs and limit their ability to adapt and experience liberty in public life.

Among the wearable assistive devices that are considered the most comfortable and useful for BVI are smart glasses, which can achieve the original goal of providing clearer vision while operating similarly to a computer. In the nearly nine years since Google announced its assistive device called "Google Glass" for BVI in 2013, many companies such as Epson, Sony, Microsoft, Envision, eSight, NuEyes, Oxsight, and OrCam have started producing smart glasses with various degrees of usability. Most such glasses have an embedded operating system and support Wi-Fi or Bluetooth for wireless communication with external devices to aid in the exploration and receiving of information in real time through the Internet, in addition to the built-in camera. Further, a touch sensor or voice recognition method may be employed as an interface to facilitate interaction between users and the smart glasses they wear. Moreover, images or video data of the surrounding environment can be obtained in real time by mounting a camera in front of the device and using computer vision methods. In the following Table 1, we compared the performance and parameters of the proposed system and other commercially available smart glass solutions for BVIs.

Table 1. The performance comparison of the proposed system and commercially available smart glasses.

Smart Glasses	Target Users	Object Recognition	Text Recognition	Independent	Tactile Graphics	Walking Night-Time	Battery Capacity
eSight 4 [19]	Low vision	No	Yes	Yes	No	No	2 h
NuEyes Pro [20]	Low vision	No	Yes	Yes	No	No	3.5 h
OrCam My Eye [21]	BVI	Yes	Yes	Yes	No	No	NA
Oxsight [22]	VI	Yes	Yes	Yes	No	No	2 h
Oton glass [23]	Low vision	No	Yes	Yes	No	No	NA
AngleEye [24]	Low vision	Yes	Yes	Yes	No	No	2 h
EyeSynth [25]	BVI	No	No	Yes	No	No	8 h
Envision [26]	BVI	Yes	Yes	Yes	No	No	5.5 h
Our System	BVI	Yes	Yes	Yes	Yes	Yes	6 h

Recently, researchers published review papers by analyzing the features of wearable assistive technologies for the BVI. Hu et al. [27] reviewed various assistive devices such as glasses, canes, gloves, and hats by analyzing behavior, structure, function, principle, context, and state of the wearable assistance system. Their analysis includes various assistive devices and 14 assistive glass research works, and 6 assistive glasses available in the market. Based on the analysis, various assistive devices can only perform on a restricted spatial scale due to their insufficient sensors and feedback methods [27]. In 2020, another survey paper on assistive technologies for BVI was published by Manjari et al. [28]. Assistive devices which were developed until 2019 were gathered and the advantages and limitations of those devices were discussed. In this year, Gupta et al. [29] studied and explored the existing assistive devices which assisted in day-to-day tasks with a simple and wearable design to give a sufficient user experience for BVI. Their findings were as follows: many assistive devices are focused on only one aspect of the problem, making it challenging for users to have a better experience; assistive devices are very heavy on a user's pocket in comparison with the features they provide, making them quite difficult to afford [29]. El-Taher et al. [30] presented a broad analysis of research relevant to assistive outdoor navigation along with commercial and non-commercial navigation applications for BVI from 2015 until 2020. One of their findings related to our smart glass system is that camera-based systems are affected by illumination and weather conditions; however, they provide more features around barriers such as shape and color.

The use of machine learning and object detection and recognition methods based on deep learning models ensures that the results obtained based on the received data from the users are reliable. In addition, before applying object recognition, the use of preprocessing methods such as contrast enhancement and noise removal is crucial, particularly in the case of low-light and dark images. Researchers have designed several approaches for the development of smart glass systems [31–34] to aid BVI people with navigation. However, these systems have major limitations. First, most of them do not employ recently developed computer vision methods, such as deep learning, and thus, the results are not efficient and less reliable. Second, they were developed to assist BVI people in crossing a road using pedestrian signals and zebra-crossing or bollard detection and recognition methods. Third, they were created considering daytime conditions and a good environment, while night-time and low-light environments were ignored. However, the effective use of cutting-edge deep learning in low-light image enhancement and object detection and recognition methods can improve awareness of surroundings and assist the confident travel of BVI people during night-time.

In this study, we proposed a smart glass system that employs computer vision techniques and deep learning models for BVI people. It was designed based on the fact that BVI people have a desire to travel at any time of the day. The proposed system can navigate BVI people even in night-time environments and comprises three parts: (1) low-light image enhancement using a two-branch exposure-fusion structure based on a convolutional neural network (CNN) to overcome noise and image enhancement limitations [35]; (2) object recognition based on a deep encoder–decoder structure using transformers [36] to help the visually impaired navigate with audio guidance; (3) salient object detection and tactile graphics generation using a two-level nested U-structure network [37] and results of our previous work [38]. Figure 1 shows the structure of a wearable navigation system with a smart glass and a refreshable tactile display. As presented in Figure 1, the main focus of this paper is marked with a red outline, and the methods in the computer vision module are explained in detail.

The primary contribution of the proposed work is as follows:

- A fully automated smart glass system was developed for BVI people to assist in the cognition process of surrounding objects during night-time. To the best of our knowledge, existing smart glass systems do not support walking in the night-time and a low-light noise environment and cannot handle night-time problems (Table 1).

- It provides users with information regarding surrounding objects through real-time audio output. In addition, it provides additional features for users to perceive salient objects using their sense of touch through a refreshable tactile display.
- The proposed system has several advantages compared to the previously developed systems; that is, the use of deep learning models in computer vision methods, and not being limited to only object detection, and global positioning system (GPS) tracking methods using basic sensors. It has four main deep learning models: low-light image enhancement, object detection, salient object extraction, and text recognition.

The remainder of this paper is organized as follows. In Section 2, we review the literature on smart glass systems and object detection and recognition. Section 3 explores the proposed system. Sections 4 and 5 discus the experimental results and highlights certain limitations of the proposed system respectively. Finally, the conclusions are presented in Section 6 including a summary of our findings and the scope for future work.

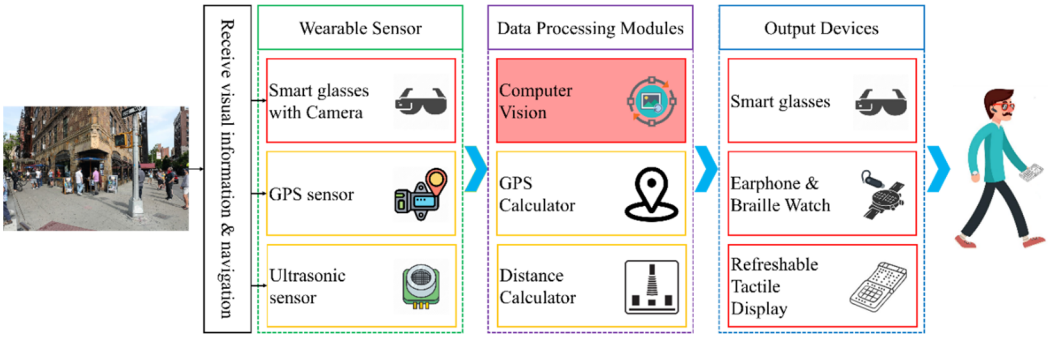

Figure 1. The structure of a wearable navigation system with smart glass and refreshable tactile display.

2. Related Works

In this section, we review studies conducted in the field of smart glass systems and object recognition. Wearable assistance systems have been developed as one of the most convenient and efficient solutions for BVI people to facilitate independent movement and performance of daily personal tasks. Smart glass systems have been employed in many fields such as health care, assisting people with visual disabilities, computer science, social science, education, service, industry, agriculture, and sports. In this literature review, we highlight the beneficial aspects of BVI people.

2.1. Smart Glass System for BVI People

One of the most important and significant tasks for BVI people is to recognize the face and identity information of relatives and friends. Daescu et al. [13] created a face recognition system that can receive facial images captured via the camera of smart glass based on commands from the user, process the result on the server, and thereafter return the result via audio. The system is designed as a client–server architecture, with a pair of cellphones, smart glasses, and a back-end server employed to implement face recognition using deep CNN models such as FaceNet and Inception-ResNet. However, this face recognition system needs to retrain to recognize new faces that are not available on the server, thereby requiring increased time to function. Mandal et al. [39] focused on the ability of recognition of faces under various lighting conditions and face poses and developed a wearable face recognition system based on Google Glasses and subclass discriminant analysis to achieve within-subclass discriminant analysis. However, this system suffers from a familiar problem; that is, although it correctly recognized the faces of 88 subjects, the model had to be retrained for new faces that were not in the initial dataset.

Further, the high price of existing commercial assistive technologies induces immense financial stress to most BVI people in developing countries and even developed countries. To solve this problem, Chen et al. [40] introduced a smart wearable system that performs

object recognition from input video frames. Their system is also built on client–server architecture, and the main image processing processes are performed on the server side, while the client side only captures images and feeds the results back to the users. As a result, the processor of the system need not employ high-priced tools, significantly reducing the cost. They used Raspberry Pi, a micro camera, and an infrared and ultrasonic sensor as the local unit, connected to the Baidu cloud server via Wi-Fi or 4G network. Furthermore, the image processing algorithm operating on the cloud server guaranteed speed and accuracy, which coupled with capturing points of interest mechanism reduced the power consumption. Ugulino and Fuks [41] described cocreation workshops and wearables prototyped by groups of BVI users, designers, mobility instructors, and computer engineering students. The group merges verbalized warnings with audio feedback and haptics to assist BVI people in recognizing landmarks. The recognition of landmarks is a necessary experience that is challenging for spatial representation and cognitive mapping. Kumar et al. [42] proposed a smart glass system to recognize objects and obstacles. It was designed with Raspberry Pi, ultrasonic sensors, mini camera, earphones, buzzer, power source, and controlled via a button to acquire photos of the surroundings concerning the user position. The primary purpose of the system was to recognize the surrounding objects using Tensorflow models and consequently alert the blind regarding collisions with obstacles via audio using ultrasonic sensors.

Traveling in large open areas and reaching the desired point poses various problems for the visually impaired because there are no tactile pavers and braille guides at such places. Consequently, Fiannaca et al. [43] proposed a navigation aid that assists BVI users using Google Glass to travel in large open areas. Their system provides secure navigation toward salient landmarks such as doors, stairs, hallway intersections, floor transitions, and water coolers by providing audio feedback to guide the BVI user towards landmarks. However, experimental results indicated that blind people typically hold the cane in their right hand to aid in navigation, which causes problems in commanding the touchpad of the smart glass using the right hand. The touchpad should be on the left side to provide a more efficient interaction with smart glass while using a cane and smart glass in parallel.

Another interesting research approach is to solve the eye contact problem of blind people in a community to facilitate conversations via eye contact with their sighted friends or partners. This problem causes feelings of social isolation and low confidence in conversations. A social glass system and tactile wrist band were implemented by Qiu et al. [44]. These two assistive devices are worn by BVI people and they assisted them in establishing eye contact and tactile feedback when eye contact was observed between blind and sighted people. Lee et al. [45] presented a concept solution to assist visually impaired people in acquiring visual information regarding pedestrians in their environment. A client and server were included in the concept solution. A server component analyzed the visual data and recognized a pedestrian based on photographs captured by the client. Face recognition, gender, age, distance calculations, and head pose are among the features available on the server. The client acquired photos and provided audio feedback to users using text-to-speech (TTS).

Furthermore, using only ultrasonic sensors in smart glass systems has also received much attention from researchers [46–48]. Hiroto and Katsumi [46] introduced a walking support system that has a glass-type wearable assistive device with an ultrasonic obstacle sensor and a pair of bone conduction earphones. Adegoke et al. [47] proposed a wearable eyeglass with an ultrasonic sensor to assist BVI people in safe navigation while avoiding objects that may be encountered, fixed, or movable, hence eliminating any potential accidents. Their system detects objects at a distance of 3–5 m, and the controller quickly alerts the user through voice feedback. However, no camera is installed to analyze the surroundings of the BVI people.

To solve the above-mentioned limitations and problems, the proposed system applied four deep learning models: low-light image enhancement, object detection, salient object extraction, and text recognition, and used the client–server architecture. The main advantages

of the proposed system over other existing systems is supporting tactile graphics generation and walking in night-time environment. Note that other existing works [13,40,45] also used a client–server architecture and increased smart glass's battery life and decreased data processing time.

2.2. Object Detection and Recognition Models

In recent years, artificial intelligence and deep learning approaches are rapidly entering all areas, including autonomous vehicle systems [49,50], robotics, space exploration, medicine, pet and animal monitoring systems [51], and areas that start with the word smart, such as smart city, smart home, smart agriculture, etc. Computer vision and artificial intelligence methods play a key role in the development of smart glass systems. It is not possible to build a smart glass system without computer vision methods such as object detection and recognition methods because the input data is an image or a video. Object detection and recognition has garnered the attention of researchers, and numerous new approaches are being developed every year. To reduce the review areas, we analyzed lightweight object detection and recognition models designed for embedded systems.

In 2016, Iandola et al. [52] designed three primary mechanisms to squeeze CNN networks and named SqueezeNet: (1) 3×3 filters were replaced with 1×1 filters; (2) the number of input channels was reduced to 3×3 filters, and (3) the network was downsampled late. These three approaches reduced the number of parameters in a CNN while maximizing the accuracy of the limited parameter sources. Further, the fire module was utilized in SqueezeNet's network architecture, which contained squeeze convolution and expansion layers. The former consists of only 1×1 convolutional filters and is fed into an expanded layer that comprises a mix of 1×1 and 3×3 convolutional filters. The output of the expanded layer is concatenated in the channel dimension such that one layer contains 1×1 convolution filters and 3×3 convolution filters. The model size achieved $50\times$ reduction compared to AlexNet and a size less than 0.5 MB was possible using the deep compression technology. Chollet [53] improved InceptionV3 by replacing a convolution with a depth-wise separable convolution and introduced the Xception model. This depth-wise separable convolution approach has been extensively applied in many other popular models such as MobileNet [54,55], ShuffleNet [56,57], and other network architectures. However, the implementation of depth-wise separable convolution is not sufficiently efficient for deep CNNs.

Mobile deep learning is rapidly expanding. The Tiny-YOLO net for iOS, introduced by Apte et al. [58] in 2017, was developed for mobile devices and tested with a metal GPU for real-time applications with an accuracy approximately similar to the original YOLO. In the same year, Howard et al. [54] built a lightweight deep neural network named MobileNet using depth-wise separable convolution architecture for mobile and embedded systems. This model has inspired researchers and has been used in various applications. In 2018, the MobileNet-SSD network [59], derived from VGG-SSD, was proposed to improve the accuracy of small objects in real-time speed. Further, Wong et al. [60] developed a compact single-shot detection deep CNN based on the remarkable performance of the fire microarchitecture presented in SqueezeNet [52] and the macro architecture introduced in SSD. A tiny SSD is created for real-time embedded systems by reducing the model size and consists of a fire subnetwork stack and optimized SSD-based convolutional feature layers. With the increasing capabilities of processors for mobile and embedded devices, numerous effective mobile deep CNNs for object detection and recognition have been introduced in recent years, such as ShuffleNet [56,57], PeleeNet [61], and EfficientDet [62].

3. The Proposed Smart Glass System

Our goal is to create convenience and opportunities for BVI people to facilitate independent travel during both day and night-time. To achieve this goal, wearable smart glass and a multifunctional system that can capture images through a mini camera and return object recognition results with voice feedback to users are the most effective approaches.

It is also conceivable to perceive visual information by touching the contours of detected salient objects according to the needs of blind people via a refreshable tactile display. The system is required to use deep CNNs to detect objects with high accuracy, and a powerful processor to perform the processes sufficiently fast in real time. Therefore, we introduced client–server architecture that consists of smart glass and a smartphone/tactile pad [63] as a local, and an artificial intelligence server to perform image processing tasks. Hereinafter, for simplicity in the text, a smartphone is written instead of a smartphone/tactile pad. The overall design of the proposed system is illustrated in Figure 2. The local part comprises smart glass and a smartphone and transfers data via a Bluetooth connection. Meanwhile, the artificial intelligence server receives the images from the local, processes them, and returns the result in audio format. Note that, smart glass hardware has a built-in speaker for direct output and earphone port for audio connection to convey returned audio results from smartphone to users.

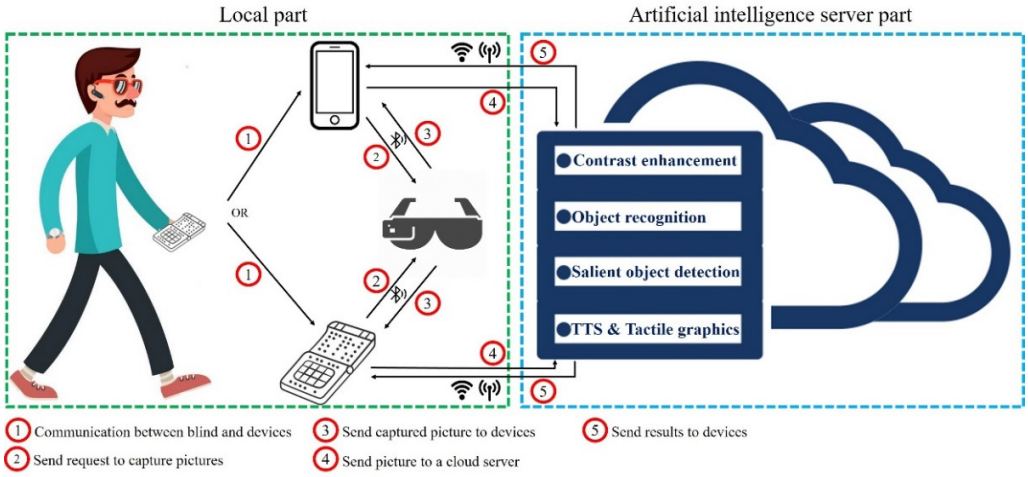

Figure 2. The overall design of the proposed system.

The working of the local part is as follows: first, the user makes a Bluetooth connection between a smart glass and a smartphone. Following this, the user can send a request to the smart glass to capture images, and the smartphone receives the images. In this scenario, the power consumption of smart glasses can be reduced, which is much more efficient than continuous video scanning. Thereafter, the results from the artificial intelligence server are delivered in voice feedback via earphones or speaker or smartphone. Further, tactile pad users can touch and sense the contours of the salient objects. Although lightweight deep CNN models have been introduced recently, we performed object detection and recognition tasks on an artificial intelligence server because the capabilities of the GPUs within wearable assistive devices and smartphones are limited compared to an artificial intelligence server. In addition, this increases the battery life of smart glasses and smartphones because they are used only for capturing images.

The artificial intelligence server part includes four main models: (1) a low-light image enhancement model, (2) an object detection and recognition model, (3) a salient object detection model, and (4) a TTS and tactile graphics generation model. Further, the artificial intelligence server part functions under two modes depending on sunrise and sunset times: daytime and night-time. In the daytime mode, the low-light image enhancement model does not function. The working of the nighttime mode is as follows (Figure 3): first, the system runs a low-light image enhancement model to increase the dark image quality and remove noise after receiving an image from a smartphone. Following the improvement in the image quality, object detection, salient object extraction, and text recognition models are applied to recognize objects, and text-to-speech is conducted. Subsequently, the audio results are returned as an artificial intelligence server response to the request made by the local. If the image is received from the tactile pad with a special title, the salient object

detection model is also performed, and the tactile graphics are also sent with the audio results as a response.

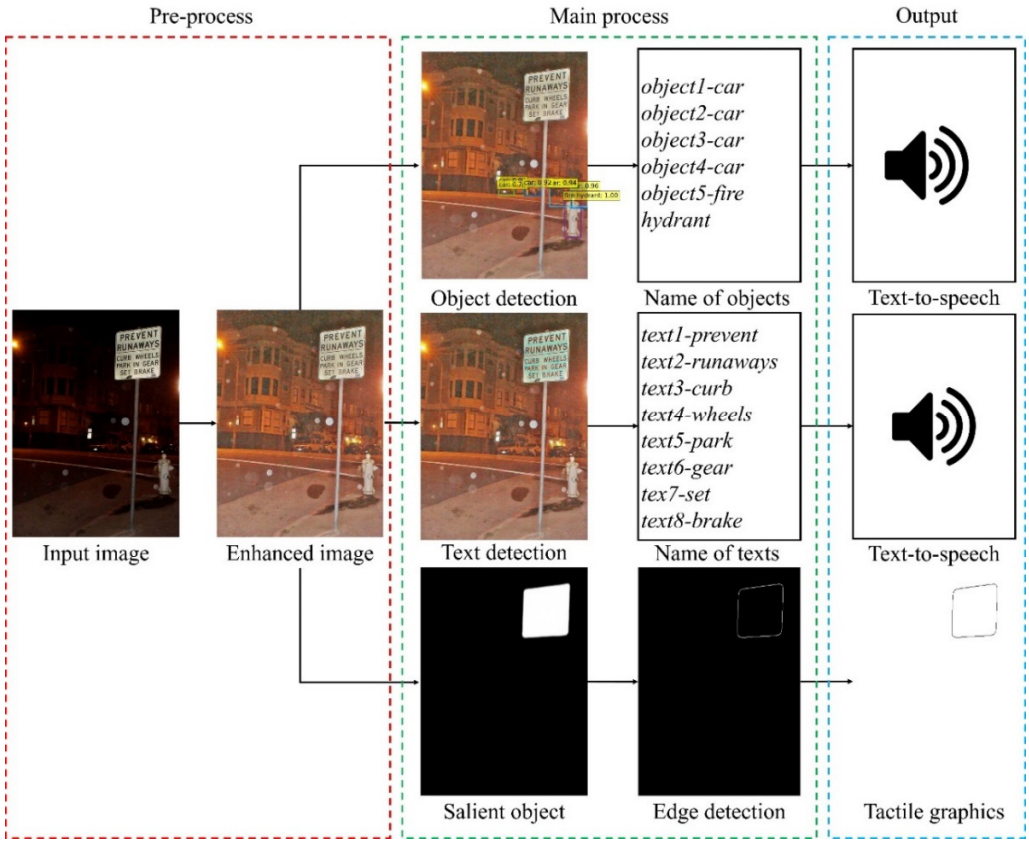

Figure 3. The main steps of the proposed system.

3.1. Low-Light Image Enhancement Model

Low-light images typically have very dark zones, blurred features, and unexpected noise, particularly when compared with well-illuminated images. This can appear when the scene is nearly dark, such as under limited luminance and night-time, or when the cameras are not set correctly. Consequently, such images show low quality owing to unsatisfactory processing of information when creating high-level applications such as object detection, recognition, and tracking owing to poor quality. Thus, this area of research is among the most valuable in computer vision, and has attracted the attention of many researchers because it is of high importance in both low-level and high-level applications such as self-driving, night vision, assistive technologies, and visual surveillance.

The use of a low-light image enhancement model for the BVI to move independently and comfortably in the dark would be an appropriate and effective solution. A low-light image enhancement model based on deep learning has recently achieved high accuracy while removing various noises. Therefore, we used a two-branch exposure-fusion network based on a CNN [35] to realize a low-light image enhancement model. A two-branch exposure-fusion network consists of two stages, wherein a two-branch illumination enhancement framework is applied in the initial step of the low-light improvement procedure, where two different enhancing approaches are employed independently to enhance the potential. A data-driven preprocessing module was presented to relieve the degradation under considerably dark conditions. Subsequently, these two enhancing modules were fed into the fusion module in the second step, which was trained to combine them with a fundamental but effective attention strategy and refining procedure. In Figure 4, we present the overall architecture of a two-branch exposure-fusion network [35]. Lu and Zhang referred to the two branches as -1E and -2E because the upper branch provides

greater support for images in the evaluation set with an exposure level of -1E, while the other branch provides greater support for images with an exposure level of -2E.

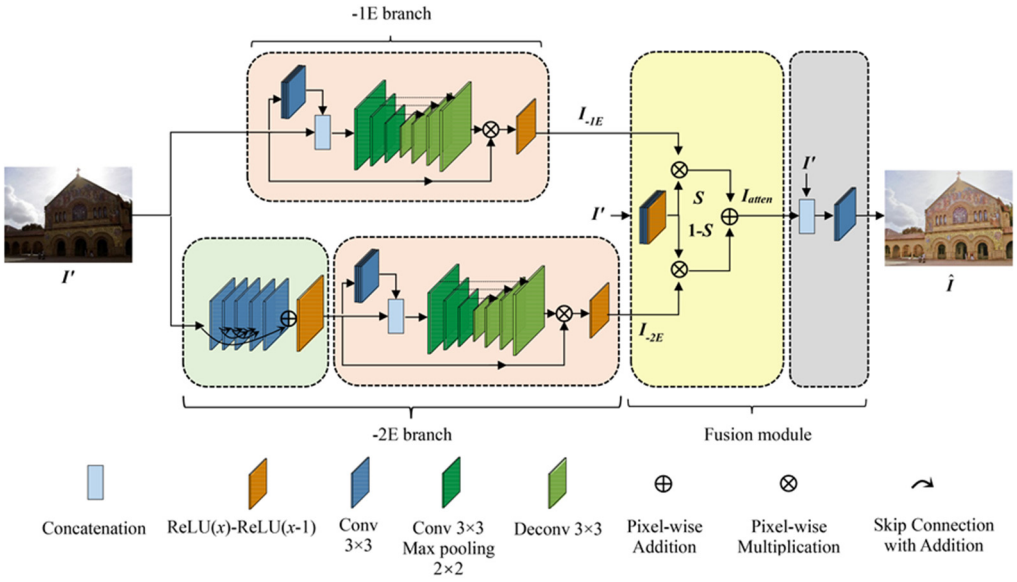

Figure 4. The network architecture of a low-light image enhancement model [35].

Basic enhancement module. F_{en}^{branch} alone constructs the -1E branch without an extra denoising method, and the main form of the -2E branch. The result of the enhancement module is represented as:

$$\boldsymbol{I}_{out}^{branch} = \boldsymbol{I}_{in}^{branch} \circ F_{en}^{branch}\left(\boldsymbol{I}_{in}^{branch}\right) \tag{1}$$

where branch $\in$ {-1E, -2E}. $\boldsymbol{I}_{in}$ and $\boldsymbol{I}_{out}$ are the input and output images, respectively. First, four convolutional layers are utilized for the input image to obtain its additional features, which are subsequently concatenated with the input low-light images before being fed into this enhancement module [35].

Preprocessing module. This module is trained in the -2E branch to separate lightly and heavily degraded images, including natural noise as the primary culprit. The preprocessing module is expressed by applying multilayer element-wise summations. Five convolutional layers with a filter size of 3×3 were applied, and their feature maps were combined with those of the previous layers to assist in the training process. Further, no activation function was implemented after the convolution layer, and only the modified *ReLU* function in the last layer was used to decrease the input properties to the range [0, 1].

$$Out(x) = ReLU(x) - ReLU(x - 1) \tag{2}$$

The range of the estimated noise was set as $(-\infty, +\infty)$ to reproduce the complex designs under low-light conditions.

Fusion module. In this module, the results enhanced by the two-branch network are first merged in the attention unit and subsequently cleaned in the refining unit to produce the final result. Four convolutional layers were applied in the attention unit to generate the attention map $\boldsymbol{S} = F_{atten}(\boldsymbol{I'})$ on the -1E enhanced image, and the equivalent element $1 - S$ for the -2E image, where $\boldsymbol{S}(x, y) \in [0, 1]$. This method aims to continuously assist in the construction of a self-adaptive fusion procedure by modifying the weighted template. The R, G, and B color channels received equal weights provided by the attention map. The results of the attention unit $\boldsymbol{I}_{atten}$ were calculated as follows:

$$\boldsymbol{I}_{atten} = \boldsymbol{I}_{\text{-1E}} \circ \boldsymbol{S} + \boldsymbol{I}_{\text{-2E}} \circ (1 - \boldsymbol{S}) \tag{3}$$

However, the disadvantage of this simple technique is that there may be a loss of certain essential features during the fusion process because the enhanced images from the -1E and -2E branches are generated independently. In addition, owing to the use of a direct metric, there may be an increase in noise. Thus, to address this, $\boldsymbol{I}_{atten}$ is sent to the refining unit $\boldsymbol{F}_{ref}$ with its low-light input concatenated. Finally, the enhanced image is formulated as:

$$\hat{\mathbf{I}} = \boldsymbol{F}_{ref}\left(concat\{\boldsymbol{I}_{atten}, \boldsymbol{I}'\}\right) \tag{4}$$

Loss Function. The combination of three loss functions such as SSIM, VGG, and Smooth was used. SSIM loss estimates the contrast, luminance, and structural diversity jointly; it is more relevant as the loss function here compared with the $L1$ and $L2$. The SSIM loss function is expressed as follows:

$$\mathcal{L}_{SSIM} = 1 - SSIM\left(\hat{\mathrm{I}},\ \mathrm{I}\right) \tag{5}$$

VGG loss is used for addressing two problems. First, when two pixels are constrained with pixel-level distance, one pixel may take the value of any pixels inside the error radius, meaning that this restriction is actually tolerant of possible shifts in the colors and color depth as stated in [35]. Second, since the ground truth is obtained using a mixture of various off-the-shelf enhancement methods, pixel-level loss functions cannot represent the desired quality correctly. It can be formulated as:

$$\mathcal{L}_{VGG} = \frac{1}{WHC}\left\|\mathcal{F}_{VGG}\left(\hat{\mathrm{I}}\right) - \mathcal{F}_{VGG}(\mathrm{I})\right\|^2 \tag{6}$$

where W, H, and C indicate the three dimensions of an image, respectively. The mean squared error was utilized to measure the distance between these features.

Smooth loss can also use total variation loss to describe both the structural features and the smoothness of the estimated transfer function, which is

$$\mathcal{L}_{SMOOTH} = \sum_{branch\{-1E,-2E\}} \left\|\nabla_{x,y}\mathcal{F}_{en}^{branch}(\mathrm{I}_{in})\right\| \tag{7}$$

where $\nabla_{x,y}$ denotes horizontal and vertical per-pixel difference. The combination of these above three loss functions are expressed as:

$$\mathcal{L} = \mathcal{L}_{SSIM} + \lambda_{vl}\cdot\mathcal{L}_{VGG} + \lambda_{sl}\cdot\mathcal{L}_{SMOOTH} \tag{8}$$

Training Data. The low-light image enhancement model was trained using Cai et al. [64] and Low-Light (LOL) datasets [65]. The value of the λ_{vl} was set to zero during the training of the -1E and -2E branches and increased to 0.1 in the joint training stage while λ_{sl} was set to 0.1 as a constant during all training. All Cai and LOL datasets were divided into training set and evaluation set. Cai dataset's images were scaled to one-fifth of the original size and then 10 patches of 256 $\times$ 256 were randomly cropped for the underexposure images of each scene. LOL dataset's images were cropped three patches for each of the images. Finally, the experiments were carried out with combination of 14,531 patches from the Cai dataset and 1449 patches from the LOL dataset.

Figure 5 shows an example of a low-light image enhancement model. The results obtained from the low-light image enhancement model were further fed into the object detection and recognition model.

Figure 5. The results of a low-light image enhancement model using a LOL dataset [65].

3.2. Object Detection and Recognition Model

To realize the object and recognition, a transformer-based encoder–decoder design [36], which is a popular design for sequence prediction, was applied. The self-attention approaches of transformers, which accurately model the interactions of elements in a sequence, render these designs particularly appropriate for collection prediction constraints, such as eliminating duplicate predictions. The Detection Transformer (DETR) predicts all objects at once and is trained end-to-end with a set loss function that achieves bipartite matching between predicted and ground-truth objects [36]. The main difference from several existing detection techniques is that DETR eliminates the need for any customized layers and thus can be regenerated simply in any structure that includes regular CNN and transformer properties. The experimental results showed that DETR achieved more reliable results for detecting large objects. However, in the case of small objects, the detection rate was lower. The network structure of the DETR is simple and is represented in Figure 6. It includes four main parts: (1) a CNN backbone to obtain a short feature description, (2) a transformer encoder, (3) a transformer decoder, and (4) a simple feedforward network (FFN) that produces the last detection prediction.

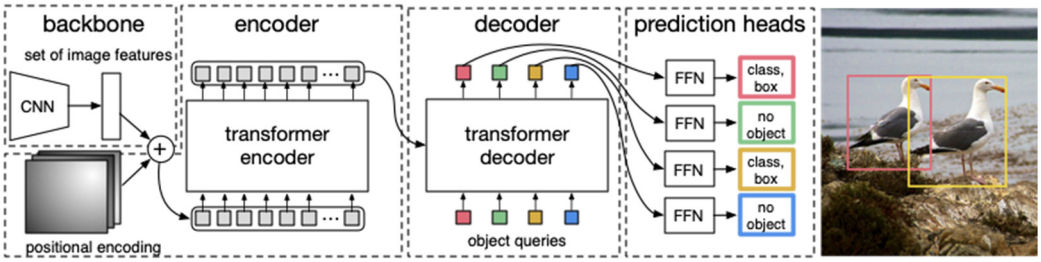

Figure 6. The network structure of the DETR [36].

Backbone. A conventional CNN backbone (ImageNet pretrained ResNet-101) produces a lower-resolution activation map $f \in \boldsymbol{R}^{C \times H \times W}$ from the input image, $x_{img} \in \boldsymbol{R}^{3 \times H_0 \times W_0}$ (with R, G, and B color channels). It is flattened and extended by the model with positional encoding before sending it into a transformer encoder.

Transformer encoder. In this section, first, the channel dimension C of the high-level activation map f is decreased to a small dimension d through a 1×1 convolution filter, and a new $z_0 \in \boldsymbol{R}^{d \times H \times W}$ feature map is created. The transformer encoder waits for the sequence as an input; therefore, the spatial dimensions of z_0 are converted to one dimension, resulting in the creation of a $d \times H \times W$ feature map. Further, each transformer encoder layer has a standard architecture and includes a multihead self-attention module and an FFN.

Transformer decoder. The decoder follows the standard structure of the transformer, converting N embeddings of size d by applying multiheaded self-attention and encoder–

decoder attention mechanisms. However, the N input embeddings must be different to create different results because the decoder is permutation-invariant. These input embeddings are determined positional encodings known as object queries, and they are added to the input of each attention layer in a manner similar to that as the encoder. Subsequently, the decoder transforms N object queries into output embedding. Thereafter, they are independently decoded via an FFN into box coordinates and class labels, producing N final predictions. The model analyzes all objects using pair-wise relationships between them by applying self-attention and encoder–decoder attention over these embeddings [36].

Prediction of Feed-Forward Networks. A three-layer perceptron with a *ReLU* activation function and hidden dimension d, as well as a linear projection layer, computes the final prediction. The normalized center coordinates, height, and width of the box with respect to the input image are predicted using the FFN, whereas the linear layer applies a *softmax* function to predict the class label. Owing to the prediction of a fixed-size set of N bounding boxes, where N is typically much larger than the actual number of objects of interest in an image, an additional special class label *NO* is utilized to indicate that no object is detected within a slot [36].

Loss Function. For auxiliary decoding losses it is convenient to use auxiliary losses [66] in the decoder during training, especially to assist the model in making the correct number of objects of each class. Prediction FFNs and Hungarian loss are added after each decoder layer.

Training Data. For training and evaluation COCO 2017 detection and panoptic segmentation datasets [67,68] are used. These datasets include 118k training images and 5k validation images. Bounding boxes and panoptic segmentation are used to label each picture. In the training set, there is an average of seven instances per image, with up to 63 occurrences in a single image, ranging in size from tiny to huge.

We experimented with an object detection and recognition model on the challenging ExDark [69] dataset. Figure 7 shows the experimental results. Subsequently, the output of the object detection and recognition model is further sent to the TTS model to generate voice feedback for blind users.

Figure 7. The results of the object detection and recognition model on the ExDark [69] dataset.

3.3. Salient Object Detection Model

We followed a two-level nested U-structure network for salient object detection [37]. Qin et al. proposed a residual U-block that includes ReSidual U-block (RSU) which has three primary components as illustrated in Figure 8: (1) an input convolution layer that converts the input feature map $x(H \times W \times C_{in})$ to an intermediate map $F_1(x)$ with a C_{out} channel, used for local feature extraction; (2) a U-Net-like symmetric encoder–decoder architecture with a height of seven that learns to extract and encode the multiscale contextual information $U(F_1(x))$ from the intermediate feature map $F_1(x)$, and (3) a residual connection that combines local features and the multiscale features via the summation $F_1(x) + U(F_1(x))$. The formula in the residual block can be summarized as $H(x) = F_2\ (F_1(x))$

+ x, where $H(x)$ indicates the desired mapping of the input features x; F_2, F_1 stand for the weight layers, which are convolution operations in this setting.

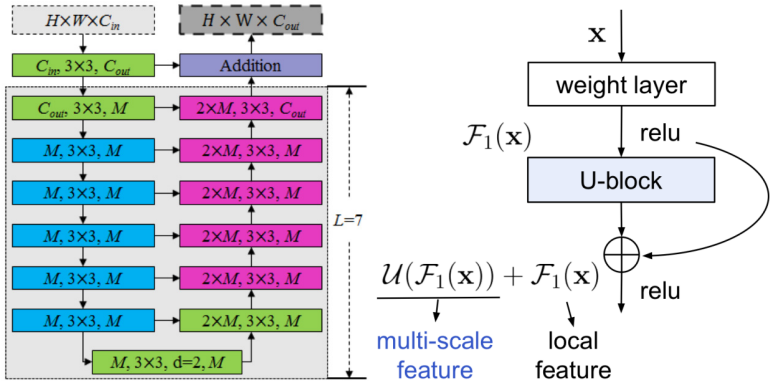

Figure 8. The structure and detail formulation of ReSidual U-block [37]. Larger L leads to deeper residual U-block.

To avoid the disadvantages of CNN network architecture with many nested, such as high computation and complexity to be employed in a real application, the two-level nested U-structure network comprised 11 stages, with each filled by a well-configured residual U-block. Further, the two-level nested U2-Net consisted of three parts: (1) a six-stage encoder, (2) a five-stage decoder, and (3) a saliency map fusion module connected to the decoder stages and the final encoder stage. The design of U2-Net was such that it supports a deep structure with rich multiscale features and has comparatively low memory costs and computation as shown in Figure 9. In encoder stages En_1, En_2, En_3, and En_4, we use residual U-blocks RSU-7, RSU-6, RSU-5, and RSU-4, respectively. As mentioned before, "7", "6", "5", and "4" denote the heights (L) of RSU blocks.

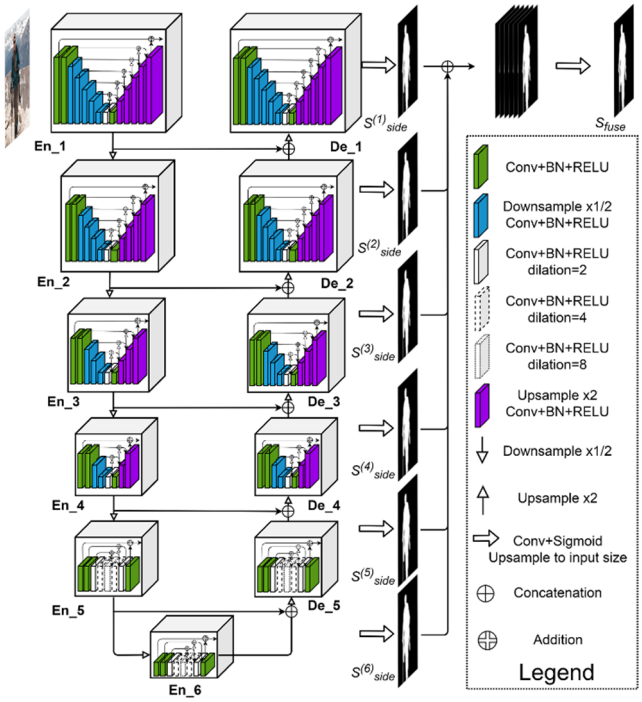

Figure 9. The network architecture of U2-Net model [37].

The decoder stages have similar arrangements to their symmetrical encoder stages concerning En_6. In De_5, the dilated version residual U-block RSU-4F was used. It is similar to encoder stages En_5 and En_6. As input, each decoder stage concatenates the

up-sampled feature maps from the previous stage with those from the symmetrical encoder stage. The saliency map fusion module, which generates saliency probability maps, is the last stage.

Furthermore, the U2-Net architecture is adaptable to a variety of working environments with minimal performance loss because it is based entirely on residual U-blocks with no reliance on any pretrained backbones adapted from image classification. The U2-Net model has versions for computers and embedded devices with sizes of 176.3 and 4.7 MB, respectively.

Training Data. For training and testing, a DUTS-TR dataset—which is a part of DUTS dataset [70]—was used. It is the most-used training dataset for salient object detection and consists of 10,553 images. To make more training images, this dataset was augmented by horizontal flipping and obtained 21,106 images.

After extracting a salient object, we can use a binary mask to obtain the contour of the salient object. These contours are used to provide visually impaired people with visual information in the form of tactile graphics. In certain situations, blind people may not be confident about objects by simply touching their contours. Therefore, we added a method to detect the inner edges of an object from images to aid in better recognition. It is necessary for a blind person to sufficiently recognize a salient object in an image and thus, we applied a binary mask to achieve the internal edges of a salient object using our previous work [38]. First, we perform a salient object by applying its binary mask by creating a matrix with a size and type similar to those of the input image to obtain the desired output image. Subsequently, we copied the non-zero pixels of the binary mask that represent the pixel of the original input image matrix as follows:

$$S_0 = B_m(x, y) * I_i(x, y) \tag{9}$$

where S_0 is the salient object, $B_m(x, y)$ is the binary mask, and $I_i(x, y)$ is the input image. Consequently, we obtained a full-color space-salient object. An example of the masking method is shown in Figure 10. Finally, we could generate the contour and inner edges of a salient object with the added helpful visual information to aid blind people in recognizing the content of an image.

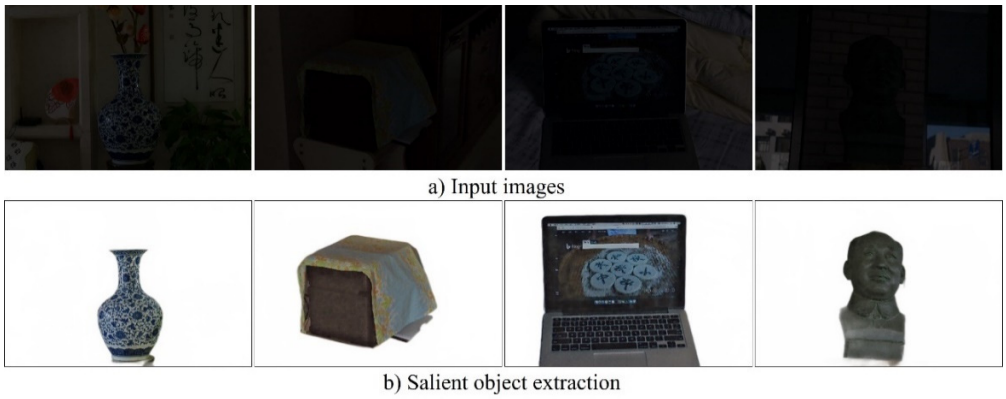

Figure 10. The results of the salient object detection model and binary masking on LOL dataset [65].

3.4. TTS and Tactile Graphics Generation Model

Blind people can receive voice feedback not only regarding surrounding objects, but also about the text data in the natural scene, which are important in our daily lives because they provide the most accurate and unambiguous descriptions of our surroundings, and can also assist blind and visually impaired people in accessing visual information. Text appears on various types of objects in natural scenes, such as billboards, road signs, and product packaging. Scene text contains valuable and high-level semantic information that is required for image comprehension; recognition can be a challenge because of variations in illumination, blurring, color differences, complex backgrounds, poor lighting

conditions, noise, and discontinuity. We used our previous real-time end-to-end scene text recognition [71] as shown in Figure 11 and Tesseract OCR engine [72] to achieve robust and accurate results on ExDark, LOL datasets, and our captured natural scene images. The fundamental part of a text detection and recognition model is a neural network model, which is trained to immediately predict the presence of text occurrences and their geometries from input images. The model is a fully convolutional network modified for text detection that results in dense per-pixel predictions of sentences or text lines. The design can be broken into three parts [71]: the feature extractor, feature merging, and the output layer. The feature extractor can be a convolutional network pretrained on the ImageNet dataset, along with interleaving convolution and pooling layers.

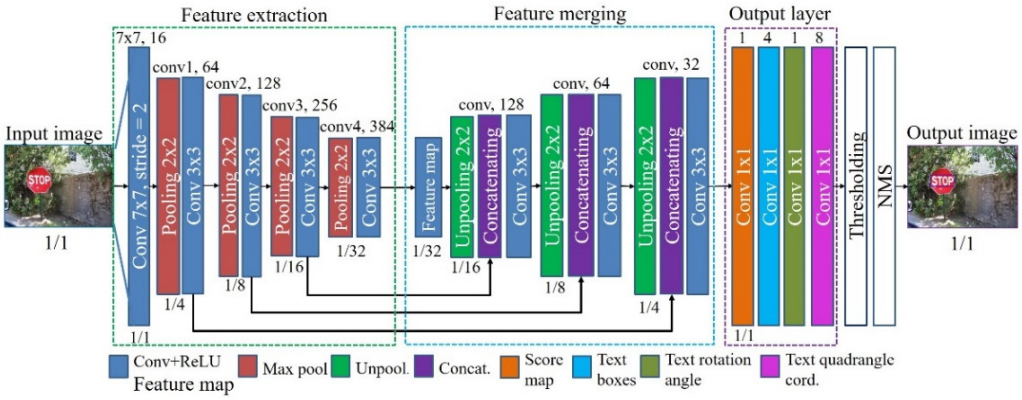

Figure 11. The network architecture of the scene text detection [71].

Texts were recognized by the trained Tesseract OCR model and sent to a TTS for pronunciation. Figure 12 shows an example of the text detection and recognition methods.

Figure 12. The results of the text detection and recognition for TTS on ExDark dataset.

Another difference between our smart glass system and other existing systems is the added function of creating tactile graphics, which provides the blind with visual information regarding the contours of salient objects. As shown in Figure 13, we created tactile graphics of salient objects using our previous work [73] and employed the tactile display system software [63] to assist the blind and visually impaired in perceiving and recognizing natural scene images.

Figure 13. The results of the tactile graphics generation on LOL dataset.

A refreshable 2D multiarray Braille display was used to dynamically represent the tactile graphics of salient objects. The tactile display has 12 × 12 Braille cells, and its simulator is illustrated in Figure 14. Further, the volume control buttons are located on the left side and can be used to adjust the volume of audio or TTS and the speed of the TTS can be increased or decreased with a long click. In addition, other buttons to control various tasks are included, as shown in Figure 14.

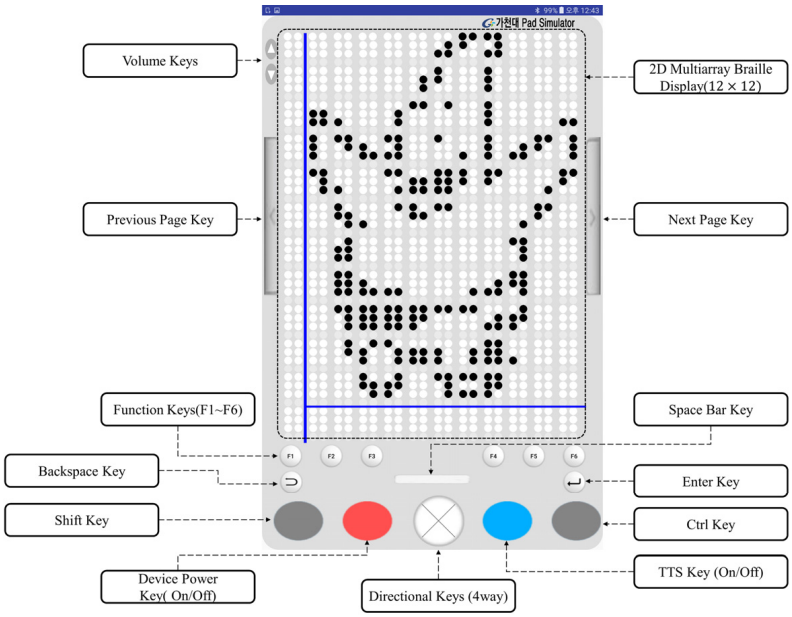

Figure 14. The design of tactile display simulator [63].

4. Experiments and Results

In this section, we present the results of the models on the artificial intelligence server. Experimental validations of the proposed smart glass system were conducted in a night-time environment, and object detection, salient object extraction, text recognition, and tactile graphics generation were focused upon. The challenging LOL dataset [65] comprising 500 low-light images and the ExDark dataset [69] comprising 7363 night images were employed. As embedded systems may not be the optimal option to increase the energy storage viability of smart glasses and ensure the real-time performance of the system, using a high-performance artificial intelligence server is more effective [74].

The performance of the artificial intelligence server determines whether the proposed smart glass system succeeds or fails. This is because the deep learning models employed in smart glass systems consume a significant amount of computing resources on an artificial intelligence server. Thus, to evaluate the performance of the proposed smart glass system, we conducted experiments using an artificial intelligence server, and the system environment is shown in Table 2.

Table 2. The environment of the artificial intelligence server.

Item	Specifications	Details
GPU	GPU 2-GeForce RTX 2080 Ti 11 GB	Two GPU are installed
CPU	Intel Core 9 Gen i7-9700k (4.90 GHz)	
RAM	DDR4 64 GB (DDR4 16GB × 4)	Samsung DDR4 16 GB PC4-21300
Storage	SSD: 512 GB/HDD: TB (2 TB × 2)	
Motherboard	ASUS PRIME Z390-A STCOM	
OS	Ubuntu Desktop	version: 18.0.4 LTS
LAN	port 1 (internal)—10/100 Mbps port 2 (external)—10/100 Mbps	
Power	1000 W (+12 V Single Rail)	Micronics Perform. II HV 1000 W Bronze

The artificial intelligence server received captured images from a local part consisting of a smartphone and smart glass. Thereafter, the received images were processed using computer vision and deep learning models. The final results were sent to the local part through Wi-Fi/Internet connection, and the user could hear the output audio information via a speaker or earphone or perceive tactile graphics using the refreshable tactile device. The experimental results of the deep learning models running on the artificial intelligence server have been presented below.

4.1. Experimental Results of Object Detection Model

First, we evaluated the performance of the object detection model, which is one of the most essential aspects of the proposed system. The object detection model was trained with AdamW [75], with initial transformer's learning rate to 10^{-4}, the backbone's to 10^{-5}, and weight decay to 10^{-4}. Before experimenting on LOL dataset, we obtained the results on COCO 2017 dataset with two varying backbones: a ResNet-50 and a ResNet-101 and compared with Faster R-CNN [76] model. The corresponding models are called, respectively, DETR-R50 and DETR-R101. In this comparison, we used an average precision (AP) metric as explained in [77]. Following [36], we also increased the feature resolution by adding a dilation to the last stage of the backbone and removing a stride from the first convolution of this stage. The corresponding models are called, respectively, DETR-DC5-R50 and DETR-DC5-R101 (dilated C5 stage). Table 3 shows a full comparison of floating point operations per second (FLOPS), frame per second (FPS), average precision (AP) of object detection with transformers (DETR), and Faster R-CNN as explained in [36].

Blind people desire to learn about the world around them during their travel, whether during daytime or night-time. Till now, object detection approaches have been efficient in environments with sufficient illumination; however, low light and a lack of illumination are among the main problems of object detection models. To address this issue, we used the low-light enhancement approach and subsequently detected objects to assist the blind user in traveling independently at any time of the day.

Table 3. The performance comparison of DETR with Faster R-CNN with ResNet-50 and ResNet-101 backbones on the COCO 2017 validation set. The results of Faster R-CNN models in Detectron2 [78] and GIoU [79] are shown in the top three rows and middle three rows, respectively. DETR models achieve comparable results to heavily tuned Faster R-CNN baselines, having lower AP_S but greatly improved AP_L. S: small objects, M: medium objects, L: large objects.

Models	GFLOPS/FPS	#Params	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
Faster RCNN-R50-DC5	320/16	166M	39.0	60.5	42.3	21.4	43.5	52.5
Faster RCNN- R50-FPN	180/26	42M	40.2	61.0	43.8	24.2	43.5	52.0
Faster RCNN-R101-FPN	246/20	60M	42.0	62.5	45.9	25.2	45.6	54.6
Faster RCNN- R50-DC5+	320/16	166M	41.1	61.4	44.3	22.9	45.9	55.0
Faster RCNN- R50-FPN+	180/26	42M	42.0	62.1	45.5	26.6	45.4	53.4
Faster RCNN-R101-FPN+	246/20	60M	44.0	63.9	**47.8**	**27.2**	48.1	56.0
DETR-R50	86/28	41M	42.0	62.4	44.2	20.5	45.8	61.1
DETR-DC5-R50	187/12	41M	43.3	63.1	45.9	22.5	47.3	61.1
DETR-R101	152/20	60M	43.5	63.8	46.4	21.9	48.0	61.8
DETR- DC5-R101	253/10	60M	**44.9**	**64.7**	47.7	23.7	**49.5**	**62.3**

We evaluated the performance of the object detection models on a low-light image following the application of the low-light enhancement method. We compared the DETR model with other 10 state-of-the-art models such as OHEM [80], Faster RCNNwFPN [81], RetinaNet [82], RefineDet512+ [83], RFBNet512-E [84], CornerNet511 [85], M2Det800 [86], R-DAD-v2 [87], ExtremeNet [88], and CenterNet511 [89]. We used the results in their papers and their source code for performance comparison. We performed quantitative analysis by using metrics such as Precision, and Recall, as in our earlier studies [38,71,90] and AP. Precision and recall rates could be obtained by comparing pixel-level ground truth images with the results of the proposed method and calculated as follows:

$$Precision_{C_{ij}} = \frac{TP_{C_{ij}}}{TP_{C_{ij}} + FP_{C_{ij}}} \tag{10}$$

$$Recall_{C_{ij}} = \frac{TP_{C_{ij}}}{TP_{C_{ij}} + FN_{C_{ij}}} \tag{11}$$

where $Precision_{C_{ij}}$ represents the Precision of category *Ci* in the *j* th image, while $Recall_{C_{ij}}$ represents the Recall of category *Ci* in the *j* th image, *TP* denotes the number of true positives indicating correctly detected object regions, *FP* denotes the number of false positives, and *FN* denotes the number of false negatives. *Precision* is defined as the number of true-positive pixels over the number of true-positive pixels plus the number of false-positive pixels. Recall is defined as the number of true-positive pixels over the number of true-positive pixels plus the number of false-negative pixels. The Average Precision (AP) of the category *Ci* can be calculated as follows:

$$AP_{C_{ij}} = \frac{1}{m} \sum_{j=1}^{m} Precision_{C_{ij}} \tag{12}$$

The comparison results of the DETR and other state-of-the-art models which are published top conferences and journals including CVVR, ICCV, ECCV, and AAAI in the recent years are presented in Table 4. As we can see, object detection with Transformers achieves the best performance on datasets LOL and ExDart in terms of AP_{50}, AP_{75}, AP_M, and AP_L evaluation metrics. DETR achieves the second-best overall performance which is slightly inferior to CenterNet511 and M2Det800 in terms of only AP and APS evaluation metrics, respectively.

Table 4. The performance comparison of DETR with other state-of-the-art methods on LOL and ExDark datasets. The best results are marked with bold. S: small objects, M: medium objects, L: large objects.

Models	Backbone Network	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
OHEM [80]	VGGNet16	21.3	40.3	21	4.3	21.5	36.8
Faster RCNN w FPN [81]	ResNet101-FPN	35.4	57.9	38.3	16.4	37.4	47.3
RetinaNet [82]	ResNeXt101-FPN	38.6	59.3	43.2	22.8	43.4	50.5
RefineDet512+ [83]	ResNet101	40.1	61.3	43.8	24.4	43.8	52.9
RFBNet512-E [84]	VGGNet16	33.1	53.2	34.7	15.9	36.1	46.3
CornerNet511 [85]	Hourglass104	40.8	55.7	43.4	19.6	43.2	55.7
M2Det800 [86]	VGGNet16	42.3	62.8	47.6	**27.5**	46.3	53.8
R-DAD-v2 [87]	ResNet101	42.7	61.6	46.8	23.5	43.6	53.2
ExtremeNet [88]	Hourglass104	41.5	59.2	46.3	22.8	45.2	55.9
CenterNet511 [89]	Hourglass104	**46**	63.1	48.5	26.8	48.3	57.2
DETR [36]	ResNet101	45.3	**63.5**	**50.3**	26.4	**48.9**	**60.7**

Figure 15 shows the results of the object detection model on the challenging LOL dataset. The experimental results indicated that in low-light images, the object detection model could correctly detect certain objects, while a few were detected incorrectly or could not be detected at all. However, more objects were correctly detected following the image illumination enhancement. The first row presents low-light images such as people, chairs, TVs, books, and different types of objects. The second and third rows display the results of the object detection model before and after the application of the low-light enhancement method, respectively.

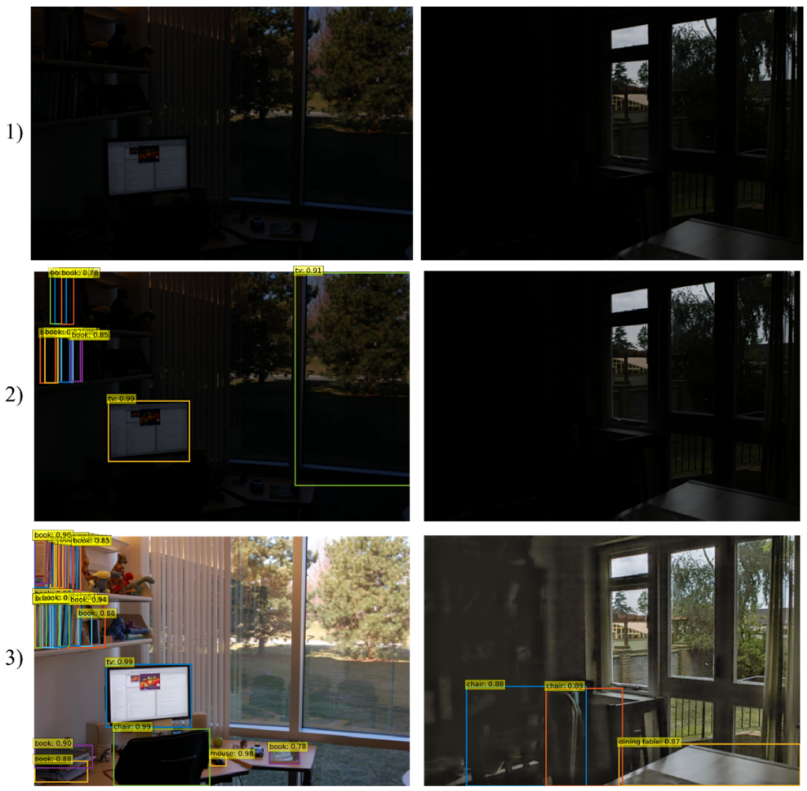

Figure 15. The results of object detection model on the challenging LOL datasets. (**1**) low-light input images; (**2**) object detection before image enhancement; (**3**) object detection after image enhancement.

Thus, the experimental results show that the object detection model performed well and accurately after image enhancement. Furthermore, it worked effectively, even when multiple objects were present, as shown in Figure 15. The data of the recognized objects were converted to audio and sent to the local part via the network.

4.2. Experimental Results of Salient Object Extraction Model

Second, we experimentally evaluated the performance of a salient object extraction model, which is one of the most significant steps in the process of creating tactile graphics from natural scene images for BVI people. Although the effective aspects and applications of salient object extraction have been emphasized by many researchers, the detection of salient objects from dark light images has not been sufficiently studied. We employed low-light image enhancement and salient object extraction models to create simple and easy-to-understand tactile graphics from low-light and dark images. As a result, BVI people could hear the name of the object around them and feel its contour via a refreshable tactile display.

To comprehensively evaluate the quality of salient object extraction methods, we additionally calculated the F-measure (FM) value, which balanced measurements between the mean of precision and recall rates and maximal F-measure (*maxFM*), weighted F-measure (*WFM*), and mean absolute error (*MAE*) metric as explained in [77]. A higher F-measure meant a higher performance and this was expressed as follows:

$$FA = \frac{(1 + 0.3) \times Precision \times Recall}{0.3 \times Precision + Recall} \tag{13}$$

A perfect match occurs when F-measure = 1 and the closer to 1 the F-measure gets, the better the detection is considered. *MAE* denotes the average per-pixel difference between a predicted saliency map and its ground truth mask. It is defined as:

$$MAE = \frac{1}{H \times W} \sum_{r=1}^{H} \sum_{c=1}^{W} |PM(r, c) - GT(r, c)| \tag{14}$$

where *PM* and *GT* are the probability map of the salient object detection and the corresponding ground truth, respectively; (*H*, *W*) and (*r*; *c*) are the (height, width) and the pixel coordinates. *WFM* is applied as a complementary measure to *maxFM* for overcoming the possible unfair comparison caused by "interpolation flaw, dependency flaw and equal-importance flaw". It is formulated as:

$$WFM = (1 + 0.3) \frac{Precision^w \times Recall^w}{0.3 \times Precision^w + Recall^w} \tag{15}$$

Table 5 shows the comparison results of three evaluation metrics and state-of-the-art performance of 10 various models which were published in top conferences such as CVVR, ICCV, and ECCV. As we can see, U2-Net obtained the best results on datasets LoL and ExDark in terms of all of the three evaluation metrics.

Further, similar to the object detection model above, the salient object extraction model first with a low-light image and subsequently after applying the low-light enhancement method were visually compared. In Figure 16, the first row shows the dark images considered, such as a flowerpot, clothes, and a microwave oven. The second row displays the salient object extraction before the low-light enhancement method. Further, the third and fourth rows show the results of the salient object extraction after the image enhancement method and salient objects in full-color space using the binary masking technique, respectively. As shown in the second row of Figure 16, the salient object extraction results from dark images exhibit incorrect extraction owing to the similar background and foreground. In contrast, the proposed salient object extraction method can reduce these drawbacks. With the help of the low-light image enhancement method, we increased the difference between the background and the object and thus efficiently extracted multiple objects.

Moreover, enhancing low-light image illumination also increases the accuracy of detecting the inner edges of salient objects using the edge detection method.

Table 5. The performance comparison of U2-Net with other state-of-the-art models on LOL and ExDark datasets. The best results are marked in bold.

Models	Backbone Network	maxFM	MAE	WFM	Published
Amulet [91]	VGGNet16	0.736	0.103	0.624	ICCV17
RAS [92]	VGGNet16	0.748	0.094	0.683	ECCV18
PiCANet [93]	VGGNet16	0.763	0.079	0.675	CVPR18
AFNet [94]	VGGNet16	0.772	0.064	0.687	CVPR19
MSWS [95]	Dense-169	0.685	0.108	0.614	CVPR19
SRM [96]	ResNet50	0.759	0.081	0.629	ICCV17
PiCANetR [93]	ResNet50	0.786	0.073	0.647	CVPR18
CPD [97]	ResNet50	0.765	0.062	0.689	CVPR19
PoolNet [98]	ResNet50	0.784	0.065	0.691	CVPR19
BASNet [99]	ResNet34	0.792	0.061	0.706	CVPR19
U2-Net [37]	RSU	**0.814**	**0.058**	**0.725**	CVPR20

Figure 16. The results of salient object extraction model on the challenging LOL datasets. (**1**) low-light input images; (**2**) salient object extraction before image enhancement; (**3**) salient object extraction after image enhancement; (**4**) salient objects in full color space.

It is essential for BVI people to fully perceive a salient object with outer and inner edges in a natural scene. Therefore, we used our previous work [38] to obtain the salient objects in a full-color space and further inner edge detection.

4.3. Experimental Results of Text-to-Speech Model

Finally, we experimentally evaluated the performance of the text-to-speech model. Text data are now encountered in all aspects of our daily lives. Therefore, conveying the text information to BVI people through audio to detect objects and convey their contours through tactile graphics is crucial. Based on these models, the BVI users can hear visual information from the natural scene around them, as shown in Figure 17.

In this study, we focused on text recognition from natural scene images in a dark environment. Because text recognition from a document or scanned images, paper documents, and books have achieved remarkable results, we used the ExDark dataset to evaluate the experimental results. We used Precision, Recall, and F-measure evaluation metrics to compare text detection and recognition models. The text detection results of our previous method and eight other cutting-edge models which were published in top conferences such as CVVR, ECCV, and AAAI are compared in Table 6.

Table 6. The performance comparison of our previous text detection model with other state-of-the-art models on ExDark datasets. The best results are marked in bold.

Methods	Backbone Network	Precision	Recall	FM	Published
Zhang et al. [100]	VGGNet16	0.705	0.414	0.527	CVPR16
Holistic [101]	VGGNet16	0.716	0.563	0.629	CVPR16
SegLink [102]	VGGNet16	0.724	0.578	0.631	CVPR17
He et al. [103]	VGGNet16	0.793	0.784	0.789	CVPR17
EAST [104]	VGGNet16	0.817	0.762	0.796	CVPR17
TextSnake [105]	VGGNet16	0.822	0.786	0.807	ECCV18
PixelLink [106]	VGGNet16	0.837	0.814	0.828	AAAI18
Wang et al. [107]	ResNet50	0.849	0.724	0.785	CVPR18
Our Previous model [71]	VGGNet16	**0.863**	**0.817**	**0.824**	IJWMIP20

The evaluation of the end-to-end system is a combination of both detection and recognition. The first predicted text examples are matched with ground truth examples after comparison of the recognized text content. The performance of end-to-end evaluation matching is initially implemented in a process similar to that of text detection. Our previous text recognition model and seven other state-of-the-art models are compared, using the ExDark dataset, in Table 7.

Table 7. The performance comparison of our previous text recognition model with other state-of-the-art models on ExDark datasets. The best results are marked in bold.

Methods	Recognition (%)	Published
Jaderberg et al. [108]	79.6	CVPR14
Shi et al. [109]	86.3	CVPR16
Shi et al. [110]	87.5	IEEE TPAMI16
Lee et al. [111]	88.2	CVPR16
Jaderberg et al. [112]	88.9	CVPR14
Shi et al. [113]	89.4	IEEE TPAMI18
Cheng et al. [114]	91.5	CVPR17
Our previous model [71]	**92.8**	IJWMIP20

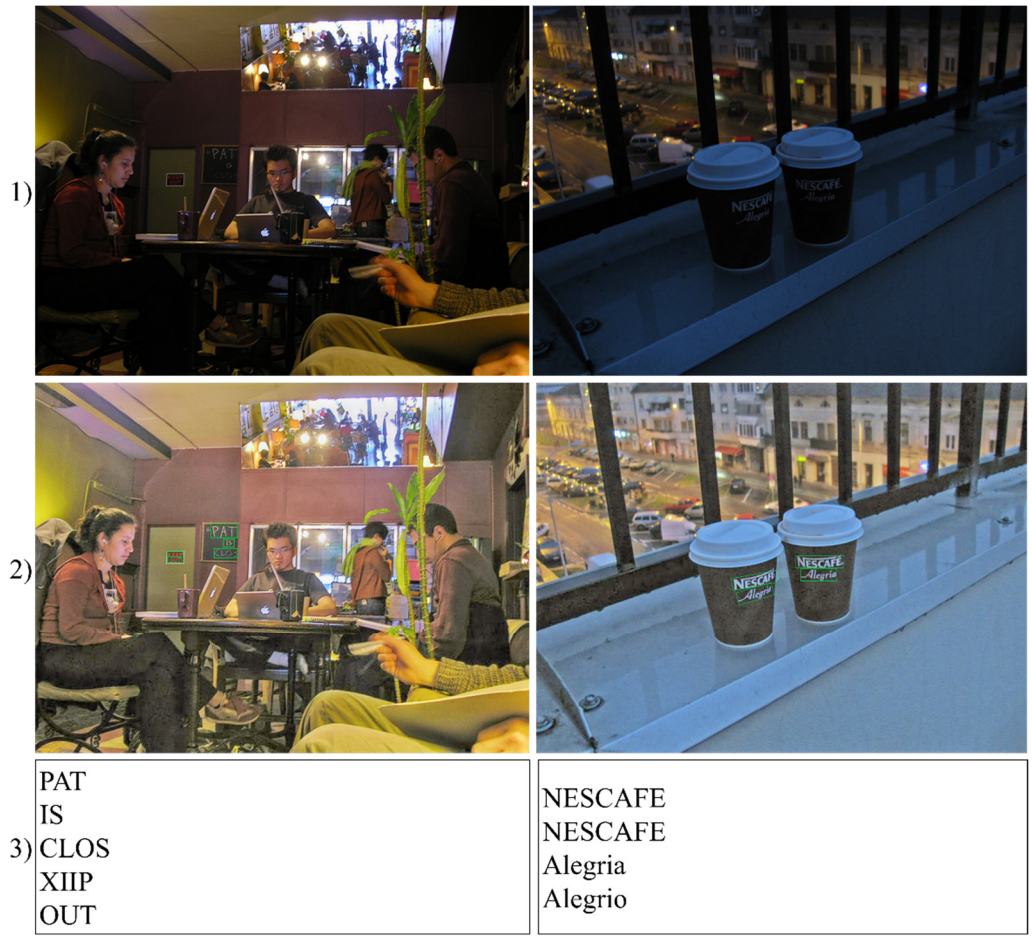

Figure 17. The results of scene text-to-speech model on the challenging ExDark datasets. (**1**) low-light input images; (**2**) text detection; (**3**) recognized text.

Figure 17 shows the results of the scene text-to-speech model obtained for the low-light images. The first row displays input images with dark scenes and different objects such as people, chairs, coffee cups, and teapots. The second and third rows show the results of the text detection method and recognized words respectively. The recognition of certain words had mistakes such as "XIIP" and "Alegrio" because of small character size and the characters being blocked by objects.

To establish the communication between client and server, we utilized gRPC (Google's Remote Procedure Call) protocol. gRPC is a free and open-source protocol that defines the bidirectional communication APIs to organize microservices between client and server. At high level (transport and application), it allows us to specify the format of REQUEST and RESPONSE messages through which the communication will be handled. gRPC protocol is built on top of HTTP/2 and inter-operates with well-known transport protocols such as TCP and UDP. It generates less latency and supports streaming, load balancing, and easy authentication procedures. At the core of gRPC, we need to define the message and services using Protocol Buffers (PB). PB efficiently serializes structured data that we call a payload and is very convenient to transport a lot of data. We also obtained the performance of frame processing time for each stage including Bluetooth image transmission between smart glass and smartphone, 5G/WiFi image transmission time between smartphone and server, and four models' image processing time in the artificial server. Table 8 presents the average processing time in seconds to perform each stage. As we can see, the total time for all stages is 0.936 s which is relevant for real-life situations.

Table 8. The performance of average frame processing time (in seconds) per sequence. The average input image size is 640 × 456.

Image Transmission and Processing	Average Processing Time (sec)
Bluetooth image transmission (between smart glass and smartphone)	0.047
5G/Wi-Fi image transmission (between smartphone and server)	0.024
Low-light image enhancement model	0.051
Object recognition model	0.173
Salient object extraction and tactile graphics model	0.215
Text recognition and TTS model	0.426
Total	**0.936**

We compared the proposed smart glass system with the other similar works in the field of wearable assistive technologies for BVI. The comparison results of the main features of different assistive systems are shown in Table 9.

Table 9. The comparison with cutting edge systems.

Systems	Image Dataset	Working Architecture	Coverage Area	Connection	Components	Results
Daescu et al. [13]	VGGFace2	Client–server	Outdoor and Indoor	5G	Smart glass, Phone, Server	Face recognition
Anandan et al. [16]	No Dataset	Local (Embedded)	Outdoor and Indoor	Offline	Raspberry-Pi, Camera, GPS	Obstacle detection
Joshi et al. [17]	Local Dataset	Local (Embedded)	Outdoor and Indoor	Offline	Distance Sensor, DSP, Camera	Object and text recognition and obstacle detection
Crose et al. [18]	No Dataset	Local (Smartphone)	Outdoor and Indoor	Offline	Smartphone, Pedestrian Dead Reckoning	Navigation
Park et al. [32]	COCO 2017, Local Dataset	Client–server	Outdoor and Indoor	NA	Raspberry-Pi, Camera	Object recognition and obstacle detection
Pardasani et al. [33]	No Dataset	Local (Embedded)	Outdoor and Indoor	Offline	Raspberry-Pi, Camera	Object and text recognition and obstacle detection
Bai et al. [34]	No Dataset	Local (Embedded)	Outdoor and Indoor	Offline	Depth Camera, Smart glass, CPU board	Obstacle detection
Mandal et al. [39]	No Dataset	Local (Google Glass)	Outdoor and Indoor	Offline	Google Glass	Face recognition
Chen et al. [40]	Labeled Faces in the Wild and PASCAL VOC	Client–server	Outdoor and Indoor	4G/Wi-Fi	Raspberry-Pi, Camera	Face, Object and text recognition
Lee et al. [45]	Local dataset	Client–server	Outdoor and Indoor	Wi-Fi	Smart glasses, phone	Face recognition
Yang et al. [115]	ADE20K, PASCAL VOC	Local (Laptop)	Outdoor and Indoor	Offline	Depth Camera, Smart glass, Laptop	Obstacle detection
Mancini et al. [116]	No Dataset	Local(Embedded)	Outdoor	Offline	Camera, PCB, and vibration motor	Obstacle detection
Patil et al. [117]	No Dataset	Local(Embedded)	Outdoor and Indoor	Offline	Sensors, Vibration motors	Obstacle detection
Al-Madani et al. [118]	No Dataset	Local(Embedded)	Indoor	Offline	BLE fingerprint, fuzzy logic	Localization in building
Our System	COCO 2017, LOL, Exdark	Client–server	Outdoor and Indoor (Nigh-time)	5G/Wi-Fi	Smart Glass, Phone, Refreshable Braille display	Object, text recognition, Tactile graphics

In addition, we obtained the experimental results using all models in the smart glass system for the sake of simplicity. The results are shown in Figure 18. The first and second columns show dark input images and the results of the image enhancement technique, respectively. The results of object detection, salient object extraction, and text detection, which are the main models of the proposed system, are shown in the third, fourth, and sixth columns, respectively. Further, the fifth column displays the results of the contour detection method used to create the tactile graphics. In the last column, recognized text from text detection is presented. The images need to be zoomed in on in order to see the specific and detailed results.

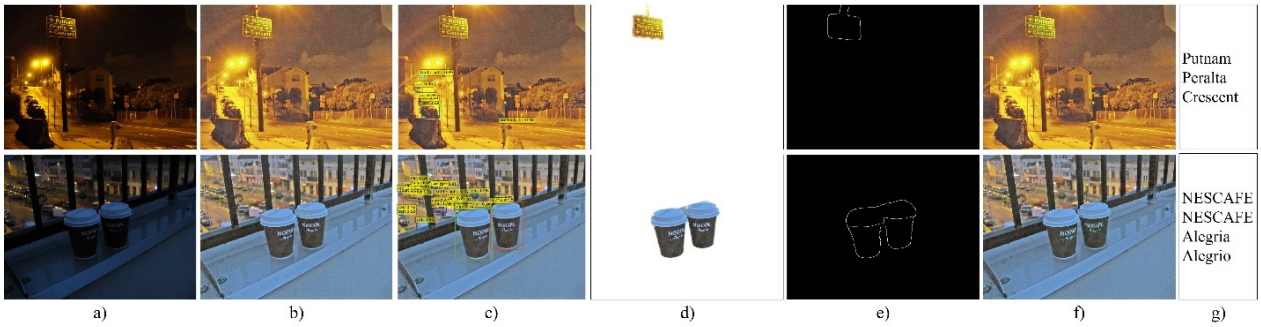

Figure 18. The result of each model of smart glass system. (**a**) Input image; (**b**) low light image enhancement; (**c**) object detection; (**d**) salient object extraction; (**e**) contour detection for tactile graphic; (**f**) text detection; (**g**) recognized text.

5. Limitation and Discussion

In addition to the aforementioned achievements, the proposed system has certain shortcomings. These drawbacks can be found in object detection, salient object extraction, and text recognition models, and experimental results with these drawbacks are shown in Figures 15–17. In certain situations, the object detection model detects more than ten objects, where a few of them are small objects or incorrectly detected, as shown in Figure 15. Further, the salient object extraction model may incorporate certain errors in extracting the regions for the cases where the image pixel values were quite close to each other, as shown in Figure 16. Furthermore, the texts were recognized from natural scene images with certain errors owing to the small size of characters, orientation, and characters being blocked by other objects, as shown in Figure 17.

Furthermore, this study covers only the artificial intelligence server part of the smart glass system and the hardware perspective that is the local part of the system and the experiments with BVI people could not be investigated owing to device patenting, pandemic, and other circumstances. We believe that in the near future, we will find solutions to these problems, conduct experiments in fully integrated software and hardware, and bring convenience to the lives of the BVI.

6. Conclusions

This paper describes a smart glass system that includes object detection, salient object extraction, and text recognition models using computer vision and deep learning for BVI people. The proposed system is fully automatic and runs on an artificial intelligence server. It detects and recognizes objects from low-light and dark-scene images to assist BVI in a night-time environment. The traditional smart glass system was extended using deep learning models and the addition of salient object extraction for tactile graphics and text recognition for text-to-speech.

Smart glass systems require greater energy and memory in embedded systems because they are based on deep learning models. Therefore, we built it in an artificial intelligence server to ensure real-time performance and solve energy problems. With the advancement of the 5G era, transmitting image data to a server or receiving real-time results for users is no longer a concern. The experimental results showed that object detection, salient object

extraction, and text recognition models were robust and performed well with the help of low-light enhancement techniques in a dark scene environment. In the future, we aim to create low-light and dark-image datasets with bounding box and ground truth data to address object detection and text recognition tasks as well as evaluations at night

Author Contributions: Conceptualization, data curation, writing—original draft, data curation, and investigation: M.M.; project administration, supervision, and writing—review and editing: J.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (2019R1F1A1057757).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Steinmetz, J.D.; Bourne, R.R.; Briant, P.S.; Flaxman, S.R.; Taylor, H.R.; Jonas, J.B.; Abdoli, A.A.; Abrha, W.A.; Abualhasan, A.; Abu-Gharbieh, E.G.; et al. Causes of blindness and vision impairment in 2020 and trends over 30 years, and prevalence of avoidable blindness in relation to VISION 2020: The Right to Sight: An analysis for the Global Burden of Disease Study. *Lancet Glob. Health* **2021**, *9*, e144–e160. [CrossRef]
2. Dunai Dunai, L.; Chillarón Pérez, M.; Peris-Fajarnés, G.; Lengua Lengua, I. Euro banknote recognition system for blind people. *Sensors* **2017**, *17*, 184. [CrossRef] [PubMed]
3. Lee, J.; Ahn, J.; Lee, K.Y. Development of a raspberry Pi-based banknote recognition system for the visually impaired. *J. Soc. E-Bus. Stud.* **2018**, *23*, 21–31.
4. Patrycja, B.-A.; Osiński, D.; Wierzchoń, M.; Konieczny, J. Visual Echolocation Concept for the Colorophone Sensory Substitution Device Using Virtual Reality. *Sensors* **2021**, *21*, 237.
5. Chang, W.-J.; Chen, L.-B.; Sie, C.-Y.; Yang, C.-H. An artificial intelligence edge computing-based assistive system for visually impaired pedestrian safety at zebra crossings. *IEEE Trans. Consum. Electron.* **2020**, *67*, 3–11. [CrossRef]
6. Yu, S.; Lee, H.; Kim, J. Street crossing aid using light-weight CNNs for the visually impaired. In Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops, Seoul, Korea, 27–28 October 2019.
7. Yuksel, B.F.; Fazli, P.; Mathur, U.; Bisht, V.; Kim, S.J.; Lee, J.J.; Jin, S.J.; Siu, Y.-T.; Miele, J.A.; Yoon, I. Human-in-the-Loop Machine Learning to Increase Video Accessibility for Visually Impaired and Blind Users. In Proceedings of the 2020 ACM Designing Interactive Systems Conference, Eindhoven, The Netherlands, 6–10 July 2020.
8. Liu, X.; Carrington, P.; Chen, X.A.; Pavel, A. What Makes Videos Accessible to Blind and Visually Impaired People? In Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems, Yokohama, Japan, 8–13 May 2021.
9. Spagnol, S.; Hoffmann, R.; Martínez, M.H.; Unnthorsson, R. Blind wayfinding with physically-based liquid sounds. *Int. J. Hum.-Comput. Stud.* **2018**, *115*, 9–19. [CrossRef]
10. Skulimowski, P.; Owczarek, M.; Radecki, A.; Bujacz, M.; Rzeszotarski, D.; Strumillo, P. Interactive sonification of U-depth images in a navigation aid for the visually impaired. *J. Multimodal User Interfaces* **2019**, *13*, 219–230. [CrossRef]
11. Zhao, Y.; Wu, S.; Reynolds, L.; Azenkot, S. A face recognition application for people with visual impairments: Understanding use beyond the lab. In Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems, Montreal, QC, Canada, 21–26 April 2018.
12. Sharma, S.; Jain, S. A static hand gesture and face recognition system for blind people. In Proceedings of the 2019 6th International Conference on Signal Processing and Integrated Networks (SPIN) IEEE, Noida, India, 7–8 March 2019.
13. Daescu, O.; Huang, H.; Weinzierl, M. Deep learning based face recognition system with smart glasses. In Proceedings of the 12th ACM International Conference on PErvasive Technologies Related to Assistive Environments, Rhodes, Greece, 5–7 June 2019.
14. Gurari, D.; Li, Q.; Lin, C.; Zhao, Y.; Guo, A.; Stangl, A.; Bigham, P.J. Vizwiz-priv: A dataset for recognizing the presence and purpose of private visual information in images taken by blind people. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–20 June 2019.
15. Rocha, D.; Carvalho, V.; Gonçalves, J.; Azevedo, F.; Oliveira, E. Development of an automatic combination system of clothing parts for blind people: MyEyes. *Sens. Transducers* **2018**, *219*, 26–33.
16. Anandan, M.; Manikandan, M.; Karthick, T. Advanced Indoor and Outdoor Navigation System for Blind People Using Raspberry-Pi. *J. Internet Technol.* **2020**, *21*, 183–195.
17. Joshi, R.C.; Yadav, S.; Dutta, M.K.; Travieso-Gonzalez, C.M. Efficient Multi-Object Detection and Smart Navigation Using Artificial Intelligence for Visually Impaired People. *Entropy* **2020**, *22*, 941. [CrossRef]
18. Croce, D.; Giarre, L.; Pascucci, F.; Tinnirello, I.; Galioto, G.E.; Garlisi, D.; Valvo, A.L. An indoor and outdoor navigation system for visually impaired people. *IEEE Access* **2019**, *7*, 170406–170418. [CrossRef]
19. eSight. Available online: https://esighteyewear.com/ (accessed on 28 October 2021).
20. NuEyes Pro. Available online: https://www.nueyes.com/ (accessed on 28 October 2021).
21. OrCam My Eye. Available online: https://www.orcam.com/en/myeye2/ (accessed on 28 October 2021).
22. Oxsight. Available online: https://oxsightglobal.com/ (accessed on 28 October 2021).

23. Oton Glass. Available online: https://www.jamesdysonaward.org/en-GB/2016/project/oton-glass/ (accessed on 28 October 2021).
24. AngleEye. Available online: https://www.closingthegap.com/angeleye-series-angleeye-smart-reader-and-angeleye-smart-glasses// (accessed on 28 October 2021).
25. EyeSynth. Available online: https://eyesynth.com/?lang=en/ (accessed on 28 October 2021).
26. Envision. Available online: https://www.letsenvision.com/envision-glasses/ (accessed on 28 October 2021).
27. Hu, M.; Chen, Y.; Zhai, G.; Gao, Z.; Fan, L. An overview of assistive devices for blind and visually impaired people. *Int. J. Robot. Autom.* **2019**, *34*, 580–598. [CrossRef]
28. Manjari, K.; Verma, M.; Singal, G. A survey on assistive technology for visually impaired. *Internet Things* **2020**, *11*, 100188. [CrossRef]
29. Gupta, L.; Varma, N.; Agrawal, S.; Verma, V.; Kalra, N.; Sharma, S. Approaches in Assistive Technology: A Survey on Existing Assistive Wearable Technology for the Visually Impaired. In *Computer Networks, Big Data and IoT*; Springer: Singapore, 2021; pp. 541–556.
30. El-Taher, F.E.Z.; Taha, A.; Courtney, J.; Mckeever, S. A systematic review of urban navigation systems for visually impaired people. *Sensors* **2021**, *21*, 3103. [CrossRef]
31. Son, H.; Krishnagiri, D.; Jeganathan, V.S.; Weiland, J. Crosswalk guidance system for the blind. In Proceedings of the 2020 42nd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC), Montreal, QC, Canada, 20–24 July 2020.
32. Park, H.; Ou, S.; Lee, J. Implementation of Multi-Object Recognition System for the Blind. *Intell. Autom. Soft Comput.* **2021**, *29*, 247–258. [CrossRef]
33. Pardasani, A.; Prithviraj, N.I.; Banerjee, S.; Kamal, A.; Garg, V. Smart assistive navigation devices for visually impaired people. In Proceedings of the 2019 IEEE 4th International Conference on Computer and Communication Systems (ICCCS), Singapore, 23–25 February 2019.
34. Jinqiang, B.; Lian, S.; Liu, Z.; Wang, K.; Liu, D. Smart guiding glasses for visually impaired people in indoor environment. *IEEE Trans. Consum. Electron.* **2017**, *63*, 258–266.
35. Lu, K.; Zhang, L. TBEFN: A two-branch exposure-fusion network for low-light image enhancement. *IEEE Trans. Multimed.* **2020**, *16*, 1–13. [CrossRef]
36. Carion, N.; Massa, F.; Synnaeve, G.; Usunier, N.; Kirillov, A.; Zagoruyko, S. End-to-end object detection with transformers. In *European Conference on Computer Vision*; Springer: Cham, Switzerland, 2020; pp. 213–229.
37. Xuebin, Q.; Zhang, Z.; Huang, C.; Dehghan, M.; Zaiane, O.R.; Jagersand, M. U2-Net: Going deeper with nested U-structure for salient object detection. *Pattern Recognit.* **2020**, *106*, 107404.
38. Mukhriddin, M.; Jeong, R.; Cho, J. Saliency cuts: Salient region extraction based on local adaptive thresholding for image information recognition of the visually impaired. *Int. Arab J. Inf. Technol.* **2020**, *17*, 713–720.
39. Bappaditya, M.; Chia, S.; Li, L.; Chandrasekhar, V.; Tan, C.; Lim, J. A wearable face recognition system on google glass for assisting social interactions. In *Asian Conference on Computer Vision*; Springer: Cham, Switzerland, 2014; pp. 419–433.
40. Shiwei, C.; Yao, D.; Cao, H.; Shen, C. A novel approach to wearable image recognition systems to aid visually impaired people. *Appl. Sci.* **2019**, *9*, 3350.
41. Ugulino, W.C.; Fuks, H. Prototyping wearables for supporting cognitive mapping by the blind: Lessons from co-creation workshops. In Proceedings of the 2015 workshop on Wearable Systems and Applications, Florence, Italy, 18 May 2015.
42. Kumar, S.N.; Varun, K.; Rahman, J.M. Object Recognition Using Perspective Glass for Blind/Visually Impaired. *J. Embed. Syst. Process* **2019**, *4*, 31–37.
43. Fiannaca, A.; Apostolopoulous, I.; Folmer, E. Headlock: A wearable navigation aid that helps blind cane users traverse large open spaces. In Proceedings of the 16th International ACM SIGACCESS Conference on Computers & Accessibility, Rochester, NY, USA, 20–22 October 2014.
44. Shi, Q.; Hu, J.; Han, T.; Osawa, H.; Rauterberg, M. An Evaluation of a Wearable Assistive Device for Augmenting Social Interactions. *IEEE Access* **2020**, *8*, 164661–164677.
45. Kyungjun, L.; Sato, D.; Asakawa, S.; Kacorri, H.; Asakawa, C. Pedestrian detection with wearable cameras for the blind: A two-way perspective. In Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems, Honolulu, HI, USA, 25–30 April 2020.
46. Kataoka, H.; Katsumi, H. A Wearable Walking Support System to provide safe direction for the Blind. In Proceedings of the 2019 34th International Technical Conference on Circuits/Systems, Computers and Communications (ITC-CSCC), Jeju, Korea, 23–26 June 2019.
47. Adegoke, A.O.; Oyeleke, O.D.; Mahmud, B.; Ajoje, J.O.; Thomase, S. Design and Construction of an Obstacle-Detecting Glasses for the Visually Impaired. *Int. J. Eng. Manuf.* **2019**, *9*, 57.
48. Ankita, B.; Laha, S.; Maity, D.K.; Sarkar, A.; Bhattacharyya, S. Smart Glass for Blind People. *AMSE J.* **2017**, *38*, 102–110.
49. Tai, S.-K.; Dewi, C.; Chen, R.-C.; Liu, Y.-T.; Jiang, X.; Yu, H. Deep Learning for Traffic Sign Recognition Based on Spatial Pyramid Pooling with Scale Analysis. *Appl. Sci.* **2020**, *10*, 6997. [CrossRef]
50. Dewi, C.; Chen, R.C.; Liu, Y.T.; Jiang, X.; Hartomo, K.D. Yolo V4 for Advanced Traffic Sign Recognition with Synthetic Training Data Generated by Various GAN. *IEEE Access* **2021**, *9*, 97228–97242. [CrossRef]

51. Chen, R.C.; Saravanarajan, V.S.; Hung, H.T. Monitoring the behaviours of pet cat based on YOLO model and raspberry Pi. *Int. J. Appl. Sci. Eng.* **2021**, *18*, 1–12. [CrossRef]
52. Iandola, F.N.; Han, S.; Moskewicz, M.W.; Ashraf, K.; Dally, W.J.; Keutzer, K. Squeezenet: Alexnet-level accuracy with 50x fewer parameters and <0.5 Mb model size. *arXiv* **2016**, arXiv:1602.07360.
53. François, C. Xception: Deep learning with depthwise separable convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017.
54. Howard, A.G.; Zhu, M.; Chen, B.; Kalenichenko, D.; Wang, W.; Weyand, T.; Andreetto, M.; Adam, H. Mobilenets: Efficient convolutional neural networks for mobile vision applications. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017.
55. Sandler, M.; Howard, A.; Zhu, M.; Zhmoginov, A.; Chen, L.-C. Mobilenetv2: Inverted residuals and linear bottlenecks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake, UT, USA, 18–22 June 2018.
56. Xiangyu, Z.; Xinyu, Z.; Mengxiao, L.; Jian, S. ShuffleNet: An extremely efficient convolutional neural network for mobile devices. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017.
57. Ma, N.; Zhang, X.; Zheng, H.-T.; Sun, J. ShuffleNet v2: Practical guidelines for efficient cnn architecture design. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018.
58. Apte, M.; Mangat, S.; Sekhar, P. YOLO Net on iOS. Available online: http://cs231n.stanford.edu/reports/2017/pdfs/135.pdf (accessed on 28 October 2021).
59. Guimei, C.; Xie, X.; Yang, W.; Liao, Q.; Shi, G.; Wu, J. Feature-fused SSD: Fast detection for small objects. In Proceedings of the Ninth International Conference on Graphic and Image Processing (ICGIP 2017), Qingdao, China, 14–16 October 2017.
60. Alexander, W.; Shafiee, M.J.; Li, F.; Chwyl, B. Tiny SSD: A tiny single-shot detection deep convolutional neural network for real-time embedded object detection. In Proceedings of the 2018 15th Conference on Computer and Robot Vision (CRV), Toronto, ON, Canada, 8–10 May 2018.
61. Wang, R.J.; Li, X.; Ling, C.X. Pelee: A real-time object detection system on mobile devices. *arXiv Prepr.* **2018**, arXiv:1804.06882.
62. Tan, M.; Pang, R.; Le, Q.V. Efficientdet: Scalable and efficient object detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, China (Virtual), 14–19 June 2020.
63. Kim, S.; Ryu, Y.; Cho, J.; Ryu, E. Towards Tangible Vision for the Visually Impaired through 2D Multiarray Braille Display. *Sensors* **2019**, *19*, 5319. [CrossRef]
64. Cai, J.; Gu, S.; Zhang, L. Learning a deep single image contrast enhancer from multi-exposure images. *IEEE Trans. Image Process.* **2018**, *27*, 2049–2062. [CrossRef] [PubMed]
65. Chen, W.; Wang, W.; Yang, W.; Liu, J. Deep retinex decomposition for low-light enhancement. *arXiv* **2018**, arXiv:1808.04560.
66. Al-Rfou, R.; Choe, D.; Constant, N.; Guo, M.; Jones, L. Character-level language modeling with deeper self-attention. In Proceedings of the AAAI Conference on Artificial Intelligence, Honolulu, HI, USA, 27 January–1 February 2019.
67. Lin, T.Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Doll'ar, P.; Zitnick, C.L. Microsoft COCO: Common objects in context. In Proceedings of the European Conference on Computer Vision (ECCV), Zurich, Switzerland, 6–12 September 2014.
68. Kirillov, A.; Girshick, R.; He, K.; Doll´ar, P. Panoptic feature pyramid networks. *arXiv* **2019**, arXiv:1901.02446.
69. Peng, L.Y.; Chan, C.S. Getting to know low-light images with the exclusively dark dataset. *Comput. Vis. Image Underst.* **2019**, *178*, 30–42.
70. Wang, L.; Lu, H.; Wang, Y.; Feng, M.; Wang, D.; Yin, B.; Ruan, X. Learning to detect salient objects with image-level supervision. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017.
71. Makhmudov, F.; Mukhiddinov, M.; Abdusalomov, A.; Avazov, K.; Khamdamov, U.; Cho, Y.I. Improvement of the end-to-end scene text recognition method for "text-to-speech" conversion. *Int. J. Wavelets Multiresolut. Inf. Process.* **2020**, *18*, 2050052-1. [CrossRef]
72. Smith, R. An overview of the Tesseract OCR engine. In Proceedings of the Ninth International Conference on Document Analysis and Recognition (ICDAR 2007), Curitiba, Brasil, 23–26 September 2007.
73. Abdusalomov, A.; Mukhiddinov, M.; Djuraev, O.; Khamdamov, U.; Whangbo, T.K. Automatic salient object extraction based on locally adaptive thresholding to generate tactile graphics. *Appl. Sci.* **2020**, *10*, 3350. [CrossRef]
74. Bai, J.; Liu, Z.; Lin, Y.; Li, Y.; Lian, S.; Liu, D. Wearable Travel Aid for Environment Perception and Navigation of Visually Impaired People. *Electronics* **2019**, *8*, 697. [CrossRef]
75. Loshchilov, I.; Hutter, F. Decoupled weight decay regularization. *arXiv* **2017**, arXiv:1711.05101.
76. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2016**, *39*, 1137–1149. [CrossRef]
77. Padilla, R.; Passos, W.L.; Dias, T.L.B.; Netto, S.L.; da Silva, E.A.B. A Comparative Analysis of Object Detection Metrics with a Companion Open-Source Toolkit. *Electronics* **2021**, *10*, 279. [CrossRef]
78. Wu, Y.; Kirillov, A.; Massa, F.; Lo, W.Y.; Girshick, R. Detectron2. 2019. Available online: https://github.com/facebookresearch/detectron2 (accessed on 28 October 2021).
79. Rezatofighi, H.; Tsoi, N.; Gwak, J.; Sadeghian, A.; Reid, I.; Savarese, S. Generalized Intersection Over Union: A Metric and a Loss for Bounding Box Regression. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–20 June 2019.

80. Shrivastava, A.; Gupta, A.; Girshick, R. Training region-based object detectors with online hard example mining. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 26 June–1 July 2016.
81. Lin, T.Y.; Dollar, P.; Girshick, R.; He, K.M.; Hariharan, B.; Belongie, S. Feature pyramid networks for object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017.
82. Lin, T.; Goyal, P.; Girshick, R.; He, K.; Dollr, P. Focal loss for dense object detection. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017.
83. Zhang, S.; Wen, L.; Bian, X.; Lei, Z.; Li, S.Z. Single-shot refinement neural network for object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake, UT, USA, 18–22 June 2018.
84. Liu, S.; Huang, D.; Wang, Y. Receptive field block net for accurate and fast object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake, UT, USA, 18–22 June 2018.
85. Law, H.; Deng, J. Cornernet: Detecting objects as paired keypoints. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018.
86. Zhao, Q.; Sheng, T.; Wang, Y.; Tang, Z.; Chen, Y.; Cai, L.; Ling, H. M2det: A single-shot object detector based on multi-level feature pyramid network. In Proceedings of the AAAI Conference on Artificial Intelligence (AAAI), Honolulu, HI, USA, 27 January–1 February 2019.
87. Bae, S.H. Object detection based on region decomposition and assembly. In Proceedings of the AAAI Conference on Artificial Intelligence (AAAI), Honolulu, HI, USA, 27 January–1 February 2019.
88. Zhou, X.; Zhuo, J.; Krahenbuhl, P. Bottom-up object detection by grouping extreme and center points. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 16–20 June 2019.
89. Duan, K.; Bai, S.; Xie, L.; Qi, H.; Huang, Q.; Tian, Q. Centernet: Keypoint triplets for object detection. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Seoul, Korea, 27 October–2 November 2019.
90. Abdusalomov, A.; Baratov, N.; Kutlimuratov, A.; Whangbo, T.K. An Improvement of the Fire Detection and Classification Method Using YOLOv3 for Surveillance Systems. *Sensors* **2021**, *21*, 6519. [CrossRef]
91. Zhang, P.; Wang, D.; Lu, H.; Wang, H.; Ruan, X. Amulet: Aggregating multi-level convolutional features for salient object detection. In Proceedings of the IEEE International Conference on Computer Vision, Honolulu, HI, USA, 21–26 July 2017.
92. Chen, S.; Tan, X.; Wang, B.; Hu, X. Reverse attention for salient object detection. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018.
93. Liu, N.; Han, J.; Yang, M. Picanet: Learning pixel-wise contextual attention for saliency detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake, UT, USA, 18–22 June 2018.
94. Feng, M.; Lu, H.; Ding, E. Attentive feedback network for boundary-aware salient object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–20 June 2019.
95. Zeng, Y.; Zhuge, Y.; Lu, H.; Zhang, L.; Qian, M.; Yu, Y. Multi-source weak supervision for saliency detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–20 June 2019.
96. Wang, T.; Borji, A.; Zhang, L.; Zhang, P.; Lu, H. A stagewise refinement model for detecting salient objects in images. In Proceedings of the IEEE International Conference on Computer Vision, Honolulu, HI, USA, 21–26 July 2017.
97. Wu, Z.; Su, L.; Huang, Q. Cascaded partial decoder for fast and accurate salient object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–20 June 2019.
98. Liu, J.; Hou, Q.; Cheng, M.; Feng, J.; Jiang, J. A simple pooling-based design for realtime salient object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–20 June 2019.
99. Qin, X.; Zhang, Z.; Huang, C.; Gao, C.; Dehghan, M.; Jagersand, M. Basnet: Boundaryaware salient object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–20 June 2019.
100. Zhang, Z.; Zhang, C.; Shen, W.; Yao, C.; Liu, W.; Bai, X. Multi-oriented text detection with fully convolutional networks. In Proceedings of the IEEE Conference Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 26 June–1 July 2016.
101. Yao, C.; Bai, X.; Sang, N.; Zhou, X.; Zhou, S.; Cao, Z. Scene text detection via holistic, multi-channel prediction. *arXiv* **2016**, arXiv:1606.09002.
102. Shi, B.; Bai, X.; Belongie, S. Detecting oriented text in natural images by linking segments. In Proceedings of the IEEE Conference Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017.
103. He, W.; Zhang, X.Y.; Yin, F.; Liu, C.L. Deep direct regression for multi-oriented scene text detection. In Proceedings of the IEEE Conference Computer Vision, Venice, Italy, 22–29 October 2017.
104. Zhou, X.; Yao, C.; Wen, H.; Wang, Y.; Zhou, S.; He, W.; Liang, J. EAST: An efficient and accurate scene text detector. In Proceedings of the IEEE Conference Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017.
105. Long, S.; Ruan, J.; Zhang, W.; He, X.; Wu, W.; Yao, C. TextSnake: A flexible representation for detecting text of arbitrary shapes. In Proceedings of the European Conference Computer Vision (ECCV), Munich, Germany, 8–14 September 2018.
106. Deng, D.; Liu, H.; Li, X.; Cai, D. Pixellink: Detecting scene text via instance segmentation. In Proceedings of the Thirty-Second AAAI Conference Artificial Intelligence, New Orleans, LA, USA, 2–7 February 2018.
107. Wang, F.; Zhao, L.; Li, X.; Wang, X.; Tao, D. Geometry-aware scene text detection with instance transformation network. In Proceedings of the IEEE Conference Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018.
108. Jaderberg, M.; Simonyan, K.; Vedaldi, A.; Zisserman, A. Deep structured output learning for unconstrained text recognition. *arXiv* **2014**, arXiv:1412.5903.

109. Shi, B.; Wang, X.; Lyu, P.; Yao, C.; Bai, X. Robust scene text recognition with automatic rectification. In Proceedings of the IEEE Conference Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 26 June–1 July 2016.
110. Shi, B.; Bai, X.; Yao, C. An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2016**, *39*, 2298–2304. [CrossRef] [PubMed]
111. Lee, C.Y.; Osindero, S. Recursive recurrent nets with attention modeling for OCR in the wild. In Proceedings of the IEEE Conference Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 26 June–1 July 2016.
112. Jaderberg, M.; Simonyan, K.; Vedaldi, A.; Zisserman, A. Reading text in the wild with convolutional neural networks. *Int. J. Comput. Vis.* **2016**, *116*, 1–20. [CrossRef]
113. Shi, B.; Yang, M.; Wang, X.; Lyu, P.; Yao, C.; Bai, X. Aster: An attentional scene text recognizer with flexible rectification. *IEEE Trans. Pattern Anal. Mach. Intell.* **2018**, *41*, 2035–2048. [CrossRef]
114. Cheng, Z.; Bai, F.; Xu, Y.; Zheng, G.; Pu, S.; Zhou, S. Focusing attention: Towards accurate text recognition in natural images. In Proceedings of the IEEE Conference Computer Vision, Venice, Italy, 22–29 October 2017.
115. Yang, K.; Bergasa, L.M.; Romera, E.; Cheng, R.; Chen, T.; Wang, K. Unifying terrain awareness through real-time semantic segmentation. In Proceedings of the 2018 IEEE Intelligent Vehicles Symposium (IV), Changshu, China, 26–30 June 2018.
116. Mancini, A.; Frontoni, E.; Zingaretti, P. Mechatronic system to help visually impaired users during walking and running. *IEEE Trans. Intell. Transp. Syst.* **2018**, *19*, 649–660. [CrossRef]
117. Patil, K.; Jawadwala, Q.; Shu, F.C. Design and construction of electronic aid for visually impaired people. *IEEE Trans. Hum.-Mach. Syst.* **2018**, *48*, 172–182. [CrossRef]
118. Al-Madani, B.; Orujov, F.; Maskeliūnas, R.; Damaševičius, R.; Venčkauskas, A. Fuzzy logic type-2 based wireless indoor localization system for navigation of visually impaired people in buildings. *Sensors* **2019**, *19*, 2114. [CrossRef]