

RGB-D and corrupted images in assistive blind systems in smart cities

Amany Yehia, Shereen A. Taie

Department of Computer Science, Faculty of Computers and Information, Fayoum University, Faiyum, Egypt

Article Info

Article history:

Received Jul 18, 2021

Revised Apr 30, 2022

Accepted Jun 1, 2022

Keywords:

Accuracy

Assertive blind system

BOBL

Object detection

Object recognition

RGB-D

YOLOv3

ABSTRACT

Assistive blind systems or assistive systems for visually impaired in smart cities help visually impaired to perform their daily tasks faced two problems when using you only look once version 3 (YOLOv3) object detection. Object recognition is a significant technique used to recognize objects with different technologies, algorithms, and structures. Object detection is a computer vision technique that identifies and locates instances of objects in images or videos. YOLOv3 is the most recent object detection technique that introduces promising results. YOLOv3 object detection task is to determine all objects, their location, and their type of objects in the scene at once so it is faster than another object detection technique. This paper solved these two problems red green blue depth (RGB-D) and corrupted images. This paper introduces two novel ways in object detection that improves YOLOv3 technique to deal with corrupted images and RGB-D images. The first phase introduces a new preprocessing model for automatically handling RGB-D on YOLOv3 with an accuracy of 61.50% in detection and 57.02% in recognition. The second phase presents a preprocessing phase to handle corrupted images to use YOLOv3 architecture with high accuracy 77.39% in detection and 71.96% in recognition.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



Corresponding Author:

Amany Yehia

Department of Computer Science, Faculty of Computers and Information, Fayoum University

Faiyum, Faiyum Governorate, Egypt

Email: ay1202@fayoum.edu.eg

1. INTRODUCTION

Assistive blind systems [1]-[4] in smart cities help visually impaired to facilitate their life such as moving from one place to another [3], [5], detect and recognize objects, find lost things and face recognition. Object recognition is a computer vision technique that recognizes, identifies, and locates objects within an image with a given degree of confidence [6]. Object recognition has many challenges, such as noise, illumination, contrast, size, angle, perspective and occlusion. The object looks different if one of its attributes changes for perspective and occlusion. Therefore, it is hard to recognize objects.

Object detection is one of computer vision techniques that identifies and locates instances of objects as humans, balls and cups in images or videos. It can verify and locate multiple objects on an image by drawing bounding boxes around every object in the image [7], [8]. These objects differ from the others in color, texture [9]. Some examples on object detection region based convolutional neural networks (R-CNN), fast R-CNN, faster R-CNN, you only look once (YOLO) and solid-state drive (SSD).

YOLO object detection is a regression-based algorithm [7]. YOLO algorithm divides image into 19*19 grid. Each cell has five bounding boxes, so the image contains 1805 bounding boxes. The bounding boxes maybe not having object. YOLO algorithm predicts the class of the object and its probability in the bounding box if there exists an object. YOLO algorithm used non-max supersession process to eliminate all

bounding boxes except the bounding boxes with the highest probability. YOLO algorithm is free and open source and very fast, so it's better to use YOLO algorithm than other object detection algorithms. There are four versions of YOLO from YOLOv1 to YOLOv4.

YOLOv3 object detection [10] is faster and more accurate than their previous work YOLOv2. YOLOv3 deals with multiple scales better. YOLOv3 algorithm also expands the network to 53 convolution layers and takes it towards residual networks by adding shortcut connections. Red green blue depth (RGB-D) image is a mixture of a RGB image and its corresponding depth image [11]. The RGB image is an image channel in which each pixel contains color and appearance information. The depth image is an image channel in which each pixel contains the distance between the image plane and the object [12]. The RGB-D image has more effective information compared with the RGB image for object recognition. In addition, the depth image is strong to variations in color and illumination. So, the RGB-D image is better performance than the RGB image for object recognition [12]. Application that uses RGB-D are assistive blind systems [4], human pose recognition [13], hand gesture recognition [14], object recognition, salient object detection [15], human action recognition [16], pedestrian detection [17], pedestrian counting systems [18], human activity detection [19] and location prediction [20].

Corrupted images are computer images that suddenly become inoperable or unusable. There are several reasons why an image may become corrupted such as motion blur, noise and camera misfocus. In some cases, it is possible to recover and fix the corrupted image. Using object detection techniques as YOLOv3 on corrupted images cause low accuracy. There are several methods that you can leverage to repair corrupted images as image enhancement and image restoration.

Image enhancement Mokhtar *et al.* [21] is the process of improving the quality of digital images so that the results can be more suitable for display or further image analysis [22]. It can remove noise, sharpen, or brighten an image, making it easier to identify key features. Image restoration is the process of converting corrupted image into original or cleaned images by eliminating or reducing degradation [23]. Bring old photos back to life (BOBL) algorithm [24] is one of image restoration algorithms. BOBL is used to improve capability of image restoration from multi defects. It contained of two phases. The first phase was a partial non-local block that targeted unstructured defects (noise, blurriness and color fading). The second phase was a local block that targeted structure defects (scratches and blotches).

The contributions of this paper are: i) introducing a new preprocessing model for detecting and recognizing the objects in the images that improve the assistive blind system in smart cities when using the YOLOv3 algorithm via introducing new preprocessing phase that enhance the performance of the YOLOv3 algorithm. Moreover, it introduces new preprocessing facilities for it to deal with RGB-D images and corrupted images; ii) the proposed model presents a new facility for the YOLOv3 algorithm to deal with RGB-D images directly through applying preprocessing tasks with an accuracy of 61.50% in detection and 57.02% in recognition; iii) the proposed model improves the results of using YOLOv3 on corrupted images through applying preprocessing phase as image restoration and image enhancement. The accuracy of YOLOv3 before applying the preprocessing phase was 61.74% in detection and 68.41% in recognition, and the accuracy was 77.39% in detection and 71.96% in recognition after applying the proposed preprocessing phase.

2. RELATED WORK

IntelliNavi navigation for blind based on kinect and machine learning (NBBKML) [3] model is a wearable navigation assistive system for the blind and the visually impaired. It detected objects that far 1-1.5 meter away from the user, and gave instructions by earphone for safe navigation. Indoor navigation system for visually impaired (INSVI) [2] is an indoor navigation system. This system estimated the user's position and his direction, and refine the orientation error using the inertial measurement unit (IMU). It recognized door numbers and detected some landmarks such as corridor corners for safe indoor navigation.

2.1. RGB-D

Large-margin multi-modal was multimodal [25], which discriminative and correlation were used. Correlation is a relation between two variables. Discriminative distinguishes boundaries between classes. This model worked on RGB and depth channels separately. They (RGB and depth) connected in the multi-modal and then applied backpropagation [26] to update the parameters of CNN. The most discriminative features for RGB and depth modality and relationship between them were discovered by multi-modal layers. The multi-modal layer could avail the complementary relationship discovered by the two modalities. Backpropagation was iteratively continued until CNN. The supervised data was back-propagated to an independent network for both color and depth from the Softmax layer. A variable was used to maximize the correlation between the two modalities. The distance between objects in the same class was small and between different classes was large. The correlation was applied by minimizing the difference for a distance of each pair (color and depth) for all objects by using canonical correlation analysis (CCA) [27]. Wang *et al.*

[25] used backpropagation to enhance performance and correlation and discriminative able to overcome on overfitting these all were the advantage of large-margin multi-modal. The convolutional and pooling layers were fixed and only fully connected layers were updated these all were the disadvantage of large-margin multi-modal. RGB-D [28] and 2D3D [29] datasets were used to learn network.

Multimodal DL for robust RGB-D [30] colorized depth image. Not only ordinary objects were recognized but also objects in noise images. This model worked on RGB and depth separately. Each stream (RGB and Depth) had a separated CNN and then gathered them together and applied the fusion network (last classification). Fine-tuned was applied to enhance accuracy for each model alone (RGB and depth). Home health aides (HHA) was applied to be effective, computationally inexpensive encoding to colorize depth image. A jet colormap was applied to colorize depth images by normalizing all depth to lie in [0-255]. Two stages to train network. First, trained the stream network that worked until layer 7 at a fully connected layer and used fixed parameter for all layers. Second, trained fusion network (fine-tuned, fusion network, Softmax). Data augmentation was used to improve recognition in noisy images. It used fine-tune and data augmentation, it worked on noisy images and its structure was simple these all were the advantages of multimodal DL for robust RGB-D [30]. It didn't use backpropagation and depth noise images worked indoor these all were the disadvantages of multimodal DL for robust RGB-D. CaffeNet [31], Washington RGB-D [28], ImageNet [32], RGB SLAM [33] and RGB-D scenes [34] datasets were used to learn network.

Semi-supervised learning model [35] built high-level features by using convolution-recursive NN (neural network). Co-training was constructed to use unlabeled RGB-D suitable for the existence of two independent views (RGB, shape or edges). As a result, it enhanced accuracy. This model worked on RGB and depth separately. Kinect was used to capture images and unsupervised feature learning. Kmean clustering was applied to learn RGB and depth modalities. A single convolutional layer was used to extract low-level features. Multiple RNN was used with fixed-tree structured to construct high level features. CNN-RNN is faster than CKM. HMP didn't need more features like surface normal. It solved large scale recognition problem and it worked on unlabeled data and high-level features for RGB and depth these all were the advantages of semi-supervised learning model [35]. The disadvantage of Semi-supervised learning was the ability of co-training was very limited. RGB-D object recognition dataset [28] was used to learn network.

RGB-D and Pose estimation OR (object recognition) [36] colorized depth images based on distance from the object center. This model enhanced accuracy by adding pose estimation in the last layer. It worked on background information. Colorizing depth used images segmentation, fill-in-holes, mesh generation, canonical perspective and finally colorized image for each depth image. Images segmentation extracted horizontal images. Fill-in-holes investigated colorization of gray-scale images. The mesh was created using a straight forward triangular. Canonical perspective created a new depth image. Pose estimation is the function of minutely localizing related parts of objects. Pose estimation was used to avoid discontinuity at an angle [0-360]. It improved state-of-the-art on the Washington dataset [36] and it improved accuracy compared with Bo *et al.* [37] these all were the advantages of RGB-D and Pose estimation OR. Comparing its accuracy with only three models this was the disadvantages of RGB-D and Pose estimation OR. Washington RGB-D [28] dataset was used to learn network.

Self-restraint [38] reduced the workload of collecting real images. It avoided gradient vanishing in the reconstruction stage and gradient exploiting in RGB. This model was divided into two stages. The first stage was the reconstruction stage, in which Auto-encoder was used. Auto-encoder added a new channel automatically to be four channels (RGB and a new image). Inserting a fully connected layers instead of the deconvolution layer in the decoding stage. The foreground was extracted from the background (objects were extracted from the image). The output of the reconstructed stage was concatenated with the original image. The second stage was multitasked learning. After concatenation, data was passed to the classification layer. The loss function is back-propagated to the auto-encoder. Pose information (triple cost) exploiting geometric information. Background images were crawled from flicker. The texture of ShapeNet [39] was used to avoid overfitting and directly trained CNN. Using batch normalization avoided overfitting. It solved the problem of characteristics of synthetic data, it solved internal covariate shift and it increased speed of convergence these all were the advantages of self-restraint [38]. It had a gap between training real images and 3D models this was the disadvantage of self-restraint. PASCAL, ImageNet and ShapeNet [39] datasets were used to learn network.

2.2. Restoration

Image inpainting for irregular holes using partial convolutions (IIHPCs) model [40] is one of the best performed inpainting methods. Partial convolutions are used on only good pixels by masking and normalizing these convolutions. This model automatically generated a new mask for the next layer. This model automatically updated mask for the next layer. IIHPCs model dealt heavily with holes of any shape, size location or distance from the image borders. Increasing holes size didn't make performance fell heavily. Generative image inpainting with contextual attention (GIICA) model [41] is one of the best and the newest

performed inpainting methods. This model improved prediction, besides collecting new image structures. Surrounding image features as references through training were used to improve prediction. GIICA model used local patch statistics and global structure. Increasing the size of the holes didn't affect on performance so this model used variable size of the holes and multiple holes with random locations.

Coherent semantic attention for image inpainting (CSAI) model [42] is one of the best-performed inpainting methods. This model introduced the attention layer to use a remote context. The coherent semantic attention (CSA) layer was used to look after contextual structure, besides effectively predicting missing parts the semantic relevance between the holes features was modeled to predict missing parts. Image inpainting via structure-aware appearance flow (IISAF) model [43] is one of the best and newest performed inpainting methods. Inpainting task was divide into two stages: structure reconstruction and texture generation. Structure reconstructor was trained by using smooth images to complete the missing structures of the input. Appearance flow was used to give details for the image in the texture generator.

3. METHOD

This paper proposes a new model for detecting and recognizing the objects in the images via introducing new preprocessing for YOLOv3 algorithm, these features enhance the performance of the YOLOv3 algorithm and introduce new abilities for it to deal with RGB-D images and Corrupted images. The proposed model compromise of two main phases, the first phase is responsible for making the YOLOv3 algorithm has the ability to deal with RGB-D images directly to detect and recognize the objects via a preprocessing task, and the second phase added the facility for YOLOv3 algorithm to deal with corrupted images with high performance via adding a preprocessing phase that increases the performance of YOLOv3 algorithm with corrupted images.

3.1. RGB-D images phase

YOLOv3 algorithm cannot deal with depth image directly. Therefore, the proposed model presents a new ability for YOLOv3 algorithm to deal with RGB-D image through adding new preprocessing phase via seven main steps as illustrated in the following subsections and represented in Figure 1. Converting image into gray, RGB-D images are received and distributed two images into two separated streams. Image is delivered to each stream and converted into a gray image to change the type of image to facilitate working on it in below steps. Each image is resized into 400*400 to have images with fixed size and facilitate the following steps.

Combining two images, these two images (RGB, Depth) are combined after converted them into gray and resized them to 400*400. Any image contains 3 channels red, green and blue. Each channel in the RGB image and the channel corresponding in depth image are taken. After that, each pixel in two channels is concatenated according to (1) and illustrated in Figure 2. In (1) where i, j is the pixel location in the channel k

$$new - pixel_{ijk} = (RGB_{ijk} + Depth_{ijk}) \% 255 \quad (1)$$

As in Figure 2, $new - pixel_{ijk} = (150 + 100) \% 255 = 250$

Enhancing, enhancement (sharpening) is applied to new image (output image from combination). Sharpness is used to make details of objects clearer. Images are enhanced with 10 % as shown in Figure 3. YOLOv3 algorithm, YOLOv3 algorithm is applied two times. The first one is on the combined image. Second one is on the enhanced image. Max, the maximum number of detected and recognized objects between two results is gotten after applying YOLOv3 on two images (enhanced image and combined image). The RGB-D layout of the proposed model can be presented as:

```

Input: RGB-D image.
Output: Detected and recognized the objects in the image.
Method:
Begin
- Step1: converting RGB image into the gray image to change its type to facilitate working on it in below steps.
- Step2: converting Depth image into the gray image to change its type to facilitate working on it in below steps.
- Step3: change the size of the two images from step1 and step2 to do a combination.
- Step4: Combining two outputs in the above two stages to make two images in one image.
- Step5: Applying enhancement (sharpening) on combined image to make objects clearer.
- Step6: applying YOLOv3 object detection on combined and enhanced images to detect and recognize objects in images.
- Step7: getting maximum between enhanced and combined images after applying YOLOv3 algorithm on each one to get the maximum number of detected and recognized objects.
End

```

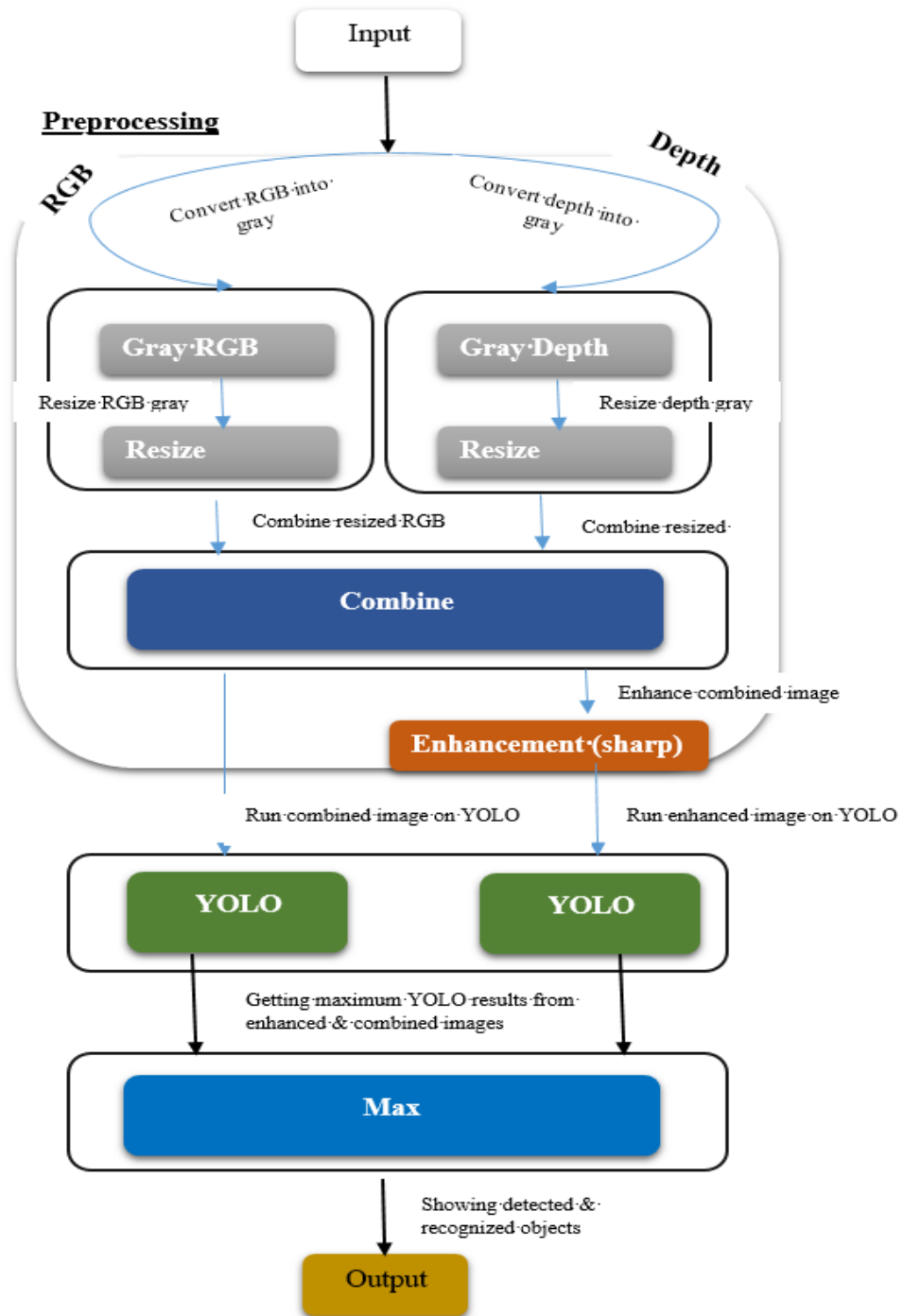


Figure 1. RGB-D framework

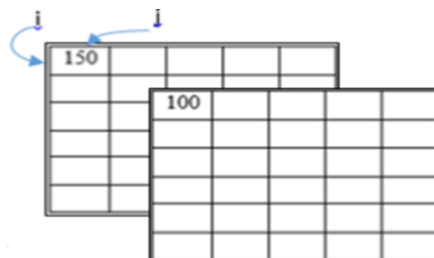


Figure 2. Channel k, i represents the number of the row and j represents the number of the column

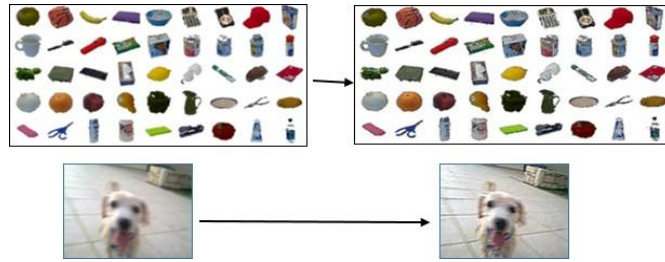


Figure 3. Results after applying enhancement on the input image

3.2. Corrupted images phase

YOLOv3 algorithm gives a low performance with corrupted images. Occasionally YOLOv3 cannot detect and recognize all objects in the image. Thus, the proposed model intended a new manner to enhance images before applying YOLOv3 object detection through a preprocessing phase with five main steps that increase the performance of the YOLOv3 algorithm with corrupted images. The proposed preprocessing phase will be illustrated in details in the following subsections and presented in Figure 4. This phase depends on enhancing the images then applied a restoration technique on them.

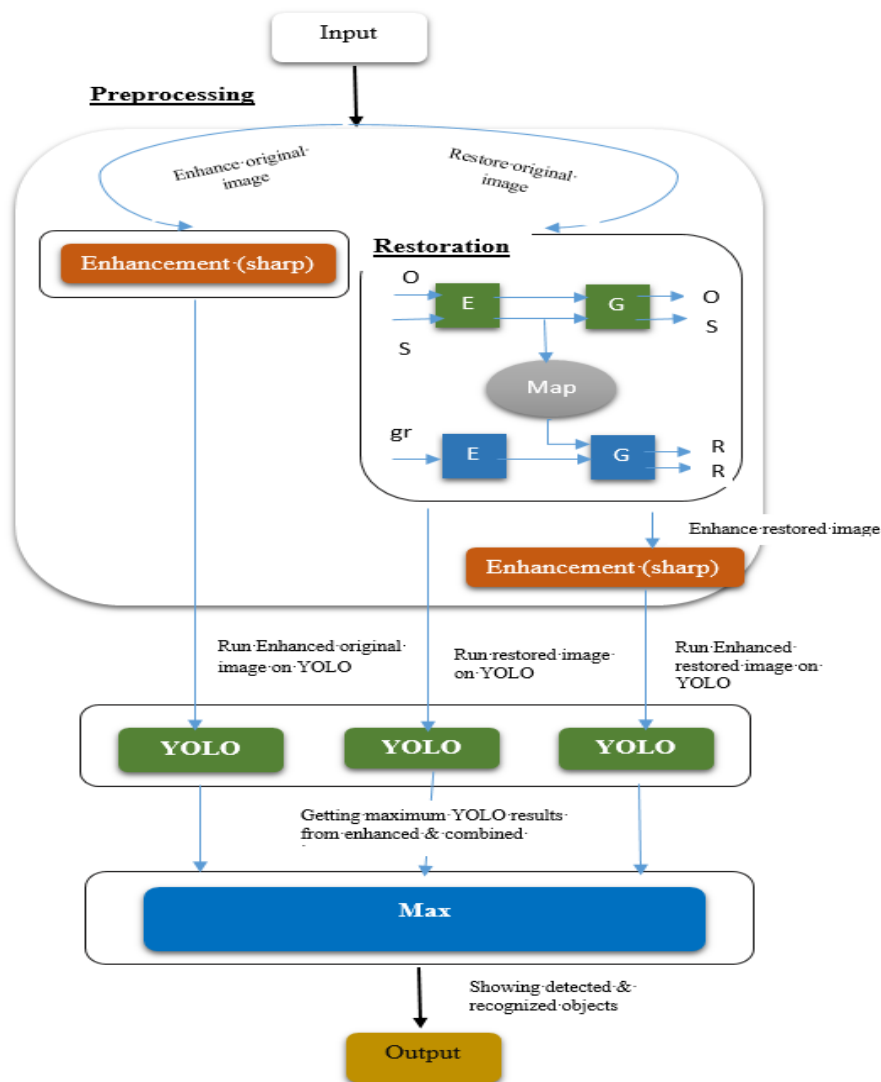


Figure 4. Corrupted image framework, E is encoder, G is generator, O is original image, S is syntactic image, gr is ground truth and R is restored image

Enhancing, enhancement (sharpening) is applied to original image. Sharpness is used to make details of objects clearer. Images are enhanced with 10% as shown in Figure 3. Restoration, BOBL used two variational auto-encoders (VAEs) to transform old photos and clean photos into two latent space and the map to restore corrupted images from latent space. VAE is designed to unsupervised learning and worked accurately in supervised and Semi-supervised learning. It can generate new images. VAEs are much more flexible and customizable in their generation behavior than GANs, so they are suitable for art generation of any kind. VAE1 had encoder and generator. Two input photos (original and synthetic image) were passed into encoder. Two outputs from encoder were delivered to generator. The output from generator is latent space for the original and synthetic image. VAE2 consist of encoder and generator. it was trained for clean images. Ground truth image was sent to encoder. The output from the encoder was passed to generator. Restored image was the output from the map (translated latent space into restored image). The results of applying BOBL as shown in Figure 5. Figure 5(a) represents the effectiveness of applying restoration in original images. Figure 5(b) is images after applying restoration.

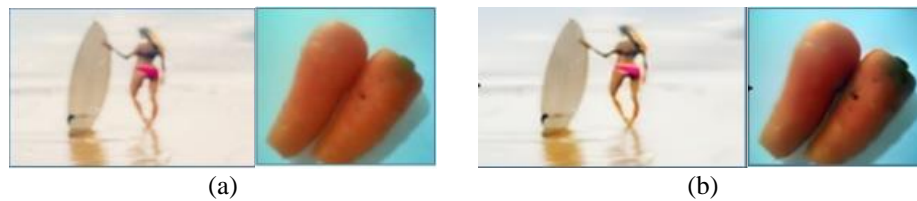


Figure 5. Effectiveness of applying restoration (a) original images and (b) after applying restoration

Enhancing, enhancement (sharpening) is applied on restored image. Images are enhanced with 10%. YOLOv3 algorithm, the proposed model applies YOLOv3 algorithm three times. First one is on enhanced original image. Second one is on restored image. Third one is on enhanced restored image.

Max, the maximum number of detected and recognized objects between three results is gotten after applying YOLOv3 on three images (enhanced original image, restored image and enhanced restored image). The layout of the proposed model can be presented as follows.

```

Input: Corrupted image or ordinary image.
Output: Detected and recognized most of objects in the image.
Method:
Begin
- Step1: Applying BOBL technique on the original image to restore corrupted images.
- Step2: Applying enhancement on the original image to prepare it for max function.
- Step3: Applying enhancement on the restored image to prepare it for max function.
- Step4: applying YOLOv3 algorithm on three output images from the above three stages to detect and recognize objects.
- Step5: getting maximum between three detected images to get the maximum number of detected and recognized objects.
End

```

4. RESULT AND DISCUSSION

4.1. RGB-D image

YOLOv3 could not automatically handle RGB-D images. The proposed model added a new ability to make YOLOv3 detect and recognize objects in RGB-D images by adding preprocessing tasks. This proposed model allowed applications that used RGB-D to used YOLOv3 to get impressive results in detection and recognition. Examples of applications that worked on RGB-D were assistive blind systems, pedestrian detection systems [15], human pose recognition systems [13] and Salient object detection [17]. This model worked on indoor and outdoor RGB-D images collected by RGB-D camera where RGB-D consist of two images one is RGB image and the other is depth image. This proposed model applied testing operation on 80 RGB-D images as shown in Figure 6. Table 1 illustrate the average results of combining RGB and Depth images (RGB-D) were 61.50% in detection and 57.02% in recognition.

Table 1. Average results of applying YOLO algorithm on RGB-D images

Proposed model (%)	
Detection	61.50
Recognition	57.02

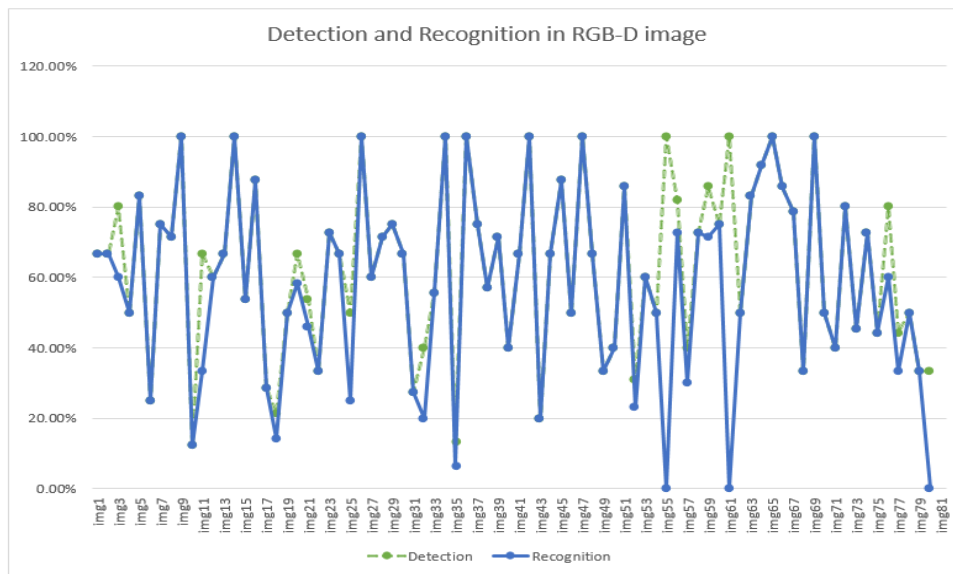


Figure 6. The results of applying the RGB-D proposed model on 80 RGB-D image

Figure 7 shows results average percentage of applying YOLOv3 on RGB-D images. Where green shape shows the average percentages of applying YOLOv3 on RGB-D whether detection or recognition. Results of applying YOLOv3 on RGB-D were 61.50% in detection and 57.02% in recognition. Figure 8 shows sample from the results of applying YOLOv3 on RGB-D images. Figures 8 (a) to (d) represent results of one RGB-D input image for each figure where image in first row and first column represents RGB image, second image in row one and column two is depth image, image in second row represents RGB image after applying YOLOv3 on it, image in third row represents combined RGB-D image after applying YOLOv3 on it. These figures illustrate the results of applying the proposed model that make more details be cleared than applying only YOLOv3. The proposed model detects and recognizes objects accurately.

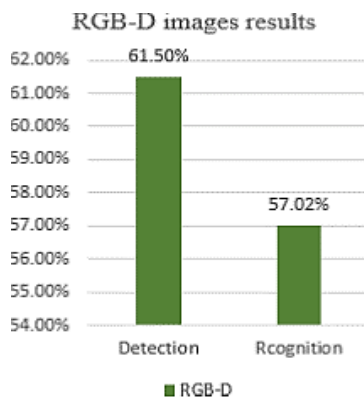


Figure 7. Results average percentage of applying YOLOv3 on RGB-D images

The proposed model added a new feature to make YOLOv3 detect and recognize objects in corrupted images with high performance to help assistive systems in smart cities through adding a preprocessing phase. The paper compared between ordinary YOLOv3 algorithm and the paper updates. This proposed model worked on is images with severe degradation. BOBL used VOC dataset [44]. This paper tested the model on 46 corrupted images as shown in Figure 9. Figure 9(a) represent the detection of 46 corrupted images. Figure 9(b) represent the recognition of 46 corrupted images. The sample results of applying the corrupted proposed model (a) blue line is the percentage of objects detection in corrupted image after applying proposed model and green dashed line is the percentage of objects detection in corrupted image after applying only YOLOv3 and (b) blue line is the percentage of objects recognition in corrupted

image after applying proposed model and green dashed line is the percentage of objects recognition in corrupted image after applying only YOLOv3.



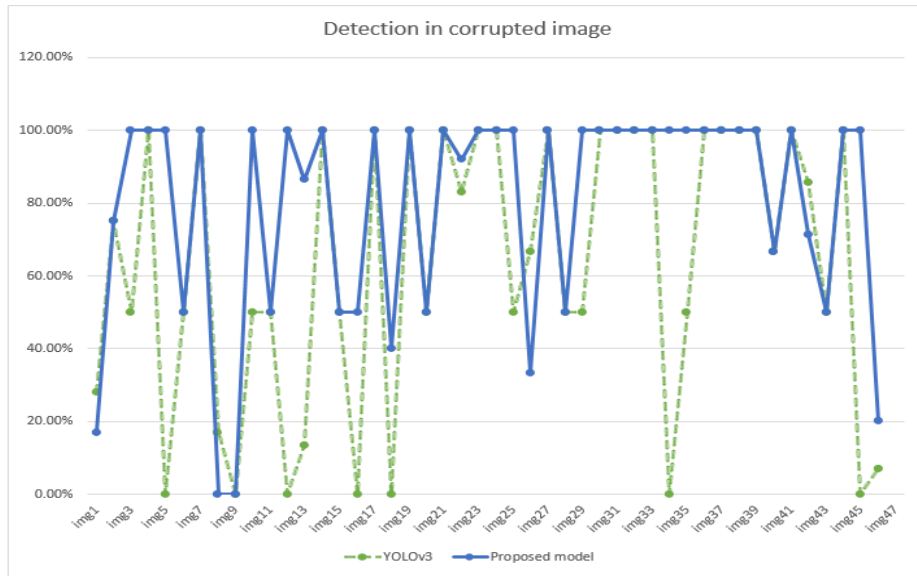
Figure 8. The sample results of applying the RGB-D proposed model where (a) is the first example, (b) is the second example, (c) is the third example and (d) is the fourth example

4.2. Corrupted image

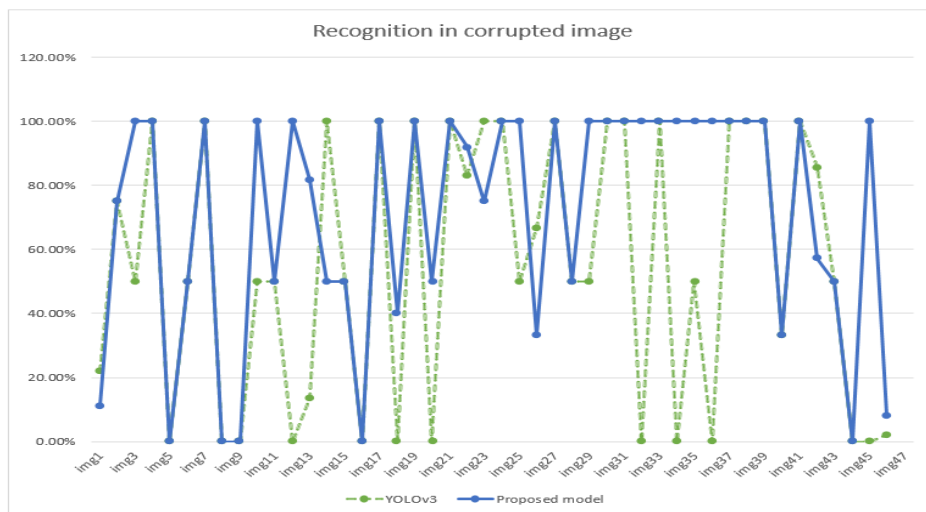
Table 2 demonstrates the average results of applying YOLOv3 algorithm on corrupted images before and after adding a new preprocessing phase to YOLOv3 algorithm. The average results of applying YOLOv3 on original corrupted images were 61.74% in detection and 53.26% in recognition. The average results of applying YOLO algorithm after adding preprocessing part on corrupted images were 77.39 % in detection and 71.96% in recognition. This part increased accuracy by 15.65% in detection and 18.7% in recognition over using ordinary YOLOv3.

Table 2. The average results of applying YOLOv3 algorithm with and without paper updates

	YOLOv3	Proposed model (%)
Detection	61.74%	77.39
Recognition	53.26%	71.96



(a)



(b)

Figure 9. Tested the model on 46 corrupted images (a) detection and (b) recognition

Figure 10 shows difference average results percentage of applying YOLOv3 on original corrupted images and corrupted images after applying preprocessing tasks that leading to increase performance. When applying YOLOv3 on original corrupted images average results were 61.74% in detection and 53.26% in recognition. Average results of applying YOLOv3 on corrupted images after applying preprocessing tasks was 77.39% in detection and 71.96% in recognition. Blue shape represents the average percentage of applying YOLOv3 on original corrupted images whether detection or recognition. Green shape shows the average percentages of applying YOLOv3 on corrupted images after applying preprocessing tasks whether detection or recognition

Figure 11 represents the sample difference between applying YOLOv3 before paper updates and YOLOv3 after paper updates. Photos were detected and recognized accurately after applying preprocessing tasks. Some photos were detected and recognized with increasing in the rate of confidence as shown in Figure 11(a). Some objects were not detected when applying YOLOv3 without using the proposed model as shown in Figure 11(b) and four column two in each row in Figure 11. When applying YOLOv3 with using proposed model, the system could detect and recognize objects well that ordinary YOLOv3 couldn't detect as shown in Figure 11(b) and four column two in each row. the proposed model could accurately recognize detected objects that ordinary YOLOv3 algorithm detect but not correctly recognize this object as shown in Figure 11(c).

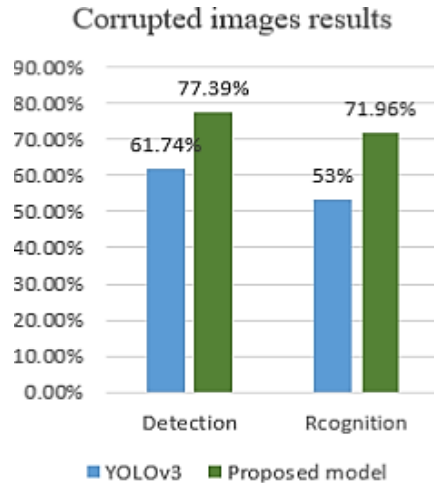


Figure 10. Difference average results percentage of applying YOLOv3

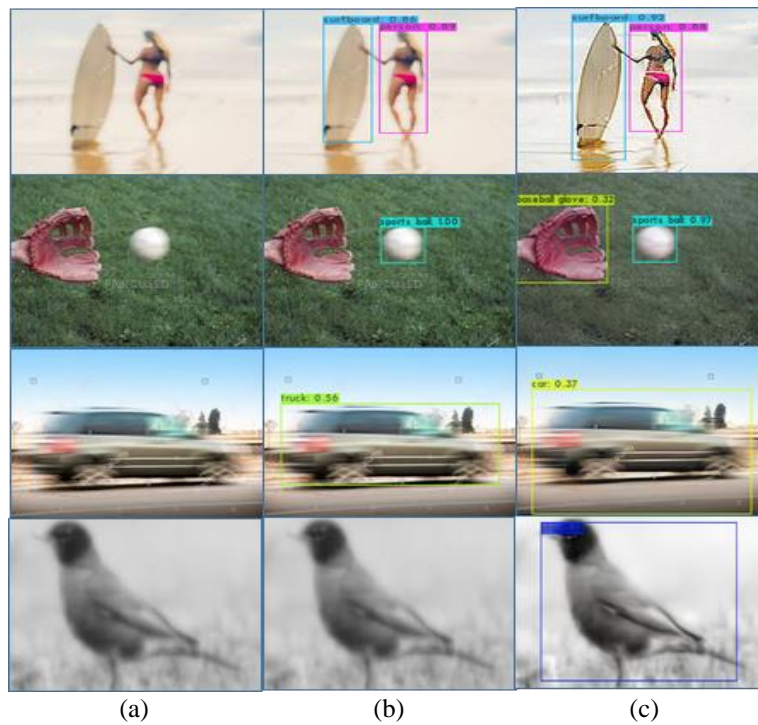


Figure 11. Sample difference between applying YOLOv3 (a) original corrupted images (b) after applying YOLOv3 and (c) corrupted images after applying preprocessing tasks and then YOLOv3

5. CONCLUSION

The assistive blind systems or visually impaired in smart cities that used RGB-D images faced two challenges when applying YOLOv3 are RGB-D and corrupted images. This paper proposes two novel models to help these systems to detect and recognize objects on RGB-D and corrupted images by improving YOLOv3 via adding preprocessing phases. When we tested a set of data on the two new models, the proposed models obtained promising results. First model adds a new feature for YOLOv3 to use RGB-D images. RGB-D architecture show this model able to handle using RGB-D on YOLOv3 with values 61.50% in detection and 57.02% in recognition. Second model adds a new feature for YOLOv3 to use Corrupted images with high performance, corrupted images architecture shows this model able to handle corrupted images on YOLOv3 by increasing performance with 15.65% in detection and 18.7% in recognition higher than using only YOLOv3 without paper updates.




REFERENCES

- [1] C. K. Lakde and P. S. Prasad, "Navigation system for visually impaired people," *2015 International Conference on Computation of Power, Energy, Information and Communication (ICCPEIC)*, 2015, pp. 0093-0098, doi: 10.1109/ICCPEIC.2015.7259447.
- [2] X. Zhang *et al.*, "A SLAM Based Semantic Indoor Navigation System for Visually Impaired Users," *2015 IEEE International Conference on Systems, Man, and Cybernetics, 2015*, pp. 1458-1463, doi: 10.1109/SMC.2015.258.
- [3] A. Bhowmick, S. Prakash, R. Bhagat, V. Prasad and S. M. Hazarika, "Intellinavi: Navigation for blind based on Kinect and machine learning," in *International Workshop on Multi-disciplinary Trends in Artificial Intelligence, Springer*, 2014, pp. 172-183, doi: 10.1007/978-3-319-13365-2_16.
- [4] Y. Tian, "RGB-D Sensor-Based Computer Vision Assistive Technology for Visually Impaired Persons," in *Computer Vision and Machine Learning with RGB-D Sensors, Springer*, 2014, pp. 173-194, doi: 10.1007/978-3-319-08651-4_9.
- [5] Y. H. Lee and G. Medioni, "Wearable rgbd indoor navigation system for the blind," in *European Conference on Computer Vision*, Springer, 2014, pp. 493-508, doi: 10.1007/978-3-319-16199-0_35.
- [6] A. Wang, J. Cai, J. Lu and T. -J. Cham, "MMSS: Multi-modal Sharable and Specific Feature Learning for RGB-D Object Recognition," *2015 IEEE International Conference on Computer Vision (ICCV)*, 2015, pp. 1125-1133, doi: 10.1109/ICCV.2015.134.
- [7] J. Redmon, S. Divvala, R. Girshick and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 779-788, doi: 10.48550/arXiv.1506.02640.
- [8] C. Szegedy, A. Toshev and D. Erhan, "Deep neural networks for object detection," *Advances in Neural Information Processing Systems*, 2013, pp. 1-9.
- [9] C. P. Papageorgiou, M. Oren and T. Poggio, "A general framework for object detection," *Sixth International Conference on Computer Vision (IEEE Cat. No.98CH36271)*, 1998, pp. 555-562, doi: 10.1109/ICCV.1998.710772.
- [10] J. Redmon and A. Farhadi, "Yolov3: An incremental improvement," *arXiv preprint arXiv: 1804.02767*, 2018, pp. 1-6, doi: 10.48550/arXiv.1804.02767.
- [11] N. Ahmed, "Spatio-temporally coherent 3d animation reconstruction from multi-view rgb-d images using landmark sampling," in *Proceedings of the International MultiConference of Engineers and Computer Scientists*, vol. 1, 2013.
- [12] H. Zeng, B. Yang, X. Wang, J. Liu, and D. Fu, "Rgb-d object recognition using multi-modal deep neural network and ds evidence theory," *Sensors*, vol. 19, no. 3, 2019, doi: 10.3390/s19030529.
- [13] J. Shotton *et al.*, "Real-time human pose recognition in parts from single depth images," *CVPR 2011*, 2011, pp. 1297-1304, doi: 10.1109/CVPR.2011.5995316.
- [14] Y. Li, X. Wang, W. Liu and B. Feng, "Deep attention network for joint hand gesture localization and recognition using static rgb-d images," *Information Sciences*, vol. 441, pp. 66-78, 2018, doi: 10.1016/j.ins.2018.02.024.
- [15] Z. Guo, W. Liao, Y. Xiao, P. Veelaert and W. Philips, "Deep learning fusion of rgb and depth images for pedestrian detection," in *30th British Machine Vision Conference*, 2019, pp. 1-13.
- [16] A. A. Chaaraoui, J. R. Padilla-Lo'pez, P. C. I Pe'rez and F. Florez-Revuelta, "Evolutionary joint selection to improve human action recognition with rgb-d devices," *Expert systems with applications*, vol. 41, no. 3, pp. 786-794, 2014, doi: 10.1016/j.eswa.2013.08.009.
- [17] Z. Liu, S. Shi, Q. Duan, W. Zhang and P. Zhao, "Salient object detection for rgb-d image by single stream recurrent convolution neural network," *Neurocomputing*, vol. 363, pp. 46-57, 2019, doi: 10.1016/j.neucom.2019.07.012.
- [18] Y. Yao, X. Zhang, Y. Liang, X. Zhang, F. Shen and J. Zhao, "A Real-Time Pedestrian Counting System Based on RGB-D," *2020 12th International Conference on Advanced Computational Intelligence (ICACI)*, 2020, pp. 110-117, doi: 10.1109/ICACI49185.2020.9177816.
- [19] J. Sung, C. Ponce, B. Selman and A. Saxena, "Unstructured human activity detection from RGBD images," *2012 IEEE International Conference on Robotics and Automation*, 2012, pp. 842-849, doi: 10.1109/ICRA.2012.6224591.
- [20] H. Hakim, Z. Alhakeem and S. Al-Darraj, "Goal location prediction based on deep learning using rgb-d camera," *Bulletin of Electrical Engineering and Informatics*, vol. 10, no. 5, 2021, doi: 10.11591/eei.v10i5.3170.
- [21] N. Mokhtar *et al.*, "Image enhancement techniques using local, global, bright, dark and partial contrast stretching for acute leukemia images," 2009, pp. 1-6.
- [22] R. Maini and H. Aggarwal, "A comprehensive review of image enhancement techniques," *arXiv preprint arXiv:1003.4053*, vol. 2, no. 3, 2010.
- [23] I. Bashir, A. Majeed and O. Khursheed, "Image restoration and the various restoration techniques used in the field of digital image processing," *International Journal of Computer Science and Mobile Computing*, vol. 6, no. 6, pp. 390-393, 2017.
- [24] Z. Wan *et al.*, "Bringing old photos back to life," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 2747-2757.
- [25] A. Wang, J. Lu, J. Cai, T. -J. Cham and G. Wang, "Large-Margin Multi-Modal Deep Learning for RGB-D Object Recognition," in *IEEE Transactions on Multimedia*, vol. 17, no. 11, pp. 1887-1898, Nov. 2015, doi: 10.1109/TMM.2015.2476655.
- [26] P. Dogra, "Rectification of corrupted neural networks," *International Journal of Science and Research (IJSR)*, vol. 5, no. 1, pp. 932-934.
- [27] G. Andrew, R. Arora, J. Bilmes and K. Livescu, "Deep canonical correlation analysis," in *International conference on machine learning*, 2013, pp. 1247-1255.
- [28] K. Lai, L. Bo, X. Ren and D. Fox, "A large-scale hierarchical multi-view RGB-D object dataset," *2011 IEEE International Conference on Robotics and Automation*, 2011, pp. 1817-1824, doi: 10.1109/ICRA.2011.5980382.
- [29] B. Browatzki, J. Fischer, B. Graf, H. H. Bülthoff and C. Wallraven, "Going into depth: Evaluating 2D and 3D cues for object classification on a new, large-scale object dataset," *2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops)*, 2011, pp. 1189-1195, doi: 10.1109/ICCVW.2011.6130385.
- [30] A. Eitel, J. T. Springenberg, L. Spinello, M. Riedmiller and W. Burgard, "Multimodal deep learning for robust RGB-D object recognition," *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2015, pp. 681-687, doi: 10.1109/IROS.2015.7353446.
- [31] Y. Jia *et al.*, "Caffe: Convolutional architecture for fast feature embedding," in *Proceedings of the 22nd ACM international conference on Multimedia*, 2014, pp. 675-678, doi: 10.1145/2647868.2654889.
- [32] O. Russakovsky *et al.*, "Imagenet large scale visual recognition challenge," *International journal of computer vision*, vol. 115, no. 3, pp. 211-252, 2015, doi: 10.1007/s11263-015-0816-y.
- [33] J. Sturm, N. Engelhard, F. Endres, W. Burgard and D. Cremers, "A benchmark for the evaluation of RGB-D SLAM systems," *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2012, pp. 573-580, doi: 10.1109/IROS.2012.6385773.

- [34] K. Lai, L. Bo, X. Ren and D. Fox, "Detection-based object labeling in 3D scenes," *2012 IEEE International Conference on Robotics and Automation*, 2012, pp. 1330-1337, doi: 10.1109/ICRA.2012.6225316.
- [35] Y. Cheng, X. Zhao, K. Huang and T. Tan, "Semi-supervised Learning for RGB-D Object Recognition," *2014 22nd International Conference on Pattern Recognition*, 2014, pp. 2377-2382, doi: 10.1109/ICPR.2014.412.
- [36] M. Schwarz, H. Schulz and S. Behnke, "RGB-D object recognition and pose estimation based on pre-trained convolutional neural network features," *2015 IEEE International Conference on Robotics and Automation (ICRA)*, 2015, pp. 1329-1335, doi: 10.1109/ICRA.2015.7139363.
- [37] L. Bo, X. Ren and D. Fox, "Unsupervised feature learning for rgb-d based object recognition," in *Experimental robotics*, Springer, 2013, pp. 387-402, doi: 10.1007/978-3-319-00065-7_27.
- [38] Y. Wang and W. Deng, "Self-restraint object recognition by model based CNN learning," *2016 IEEE International Conference on Image Processing (ICIP)*, 2016, pp. 654-658, doi: 10.1109/ICIP.2016.7532438.
- [39] A. X. Chang *et al.*, "Shapenet: An information-rich 3d model repository," *arXiv preprint arXiv:1512.03012*, 2015, doi: 10.48550/arXiv.1512.03012.
- [40] G. Liu, F. A. Reda, K. J. Shih, T.-C. Wang, A. Tao and B. Catanzaro, "Image inpainting for irregular holes using partial convolutions," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 85-100, 10.48550/arXiv.1804.07723.
- [41] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu and T. S. Huang, "Generative image inpainting with contextual attention," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 5505-5514, doi: 10.1109/CVPR.2018.00577.
- [42] H. Liu, B. Jiang, Y. Xiao and C. Yang, "Coherent semantic attention for image inpainting," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 4170-4179, doi: 10.48550/arXiv.1905.12384.
- [43] Y. Ren, X. Yu, R. Zhang, T. H. Li, S. Liu and G. Li, "Structureflow: Image inpainting via structure-aware appearance flow," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 181-190, doi: 10.48550/arXiv.1908.03852.
- [44] R. Heckenauer *et al.*, "Détection en temps réel des glomérules en pathologie rénale," in *ORASIS 2021*, 2021.

BIOGRAPHIES OF AUTHORS



Amany Yehia    received the B.Sc degree in Computer Science by faculty of Computer and information, Fayoum university, Egypt in 2017. She can be contacted at email: ay1202@fayoum.edu.eg.



Shereen A. Taie    is an associate professor in Computer Science Department, Faculty of Computers and Information, Fayoum University, Egypt, and Head of Center of Electronic Courses Production, Fayoum University, Egypt. She held an M.S. in 2006 in Computer Science from Computer and Mathematics Department Faculty of Science, Cairo University, Egypt, and a Ph.D. in 2012 in Computer Science from Computer and Mathematics Department Faculty of Science, Cairo University, Egypt. She was awarded a B.Sc. degree in Computer Science by Faculty of Science, Cairo University in 1996. She is the Vice Dean for Education and Students Affairs, Faculty of Computers and Information, Fayoum University, Egypt. She can be contacted at email: sat00@fayoum.edu.eg