International Journal of Computer Sciences and Engineering Open Access
Research Paper Vol-6, Special Issue-6, July 2018 E-ISSN: 2347-2693

Automatic Occlusion Removal System using Optical Flow Method

M.A. Aneesha[1*], K.J. Helen[2]

[1,2]Department of Computer Science and Engineering, Government Engineering College Thrissur, APJ Abdul Kalam Technological University, Thiruvananthapuram, India

[*]Corresponding Author: aneeshagec@gmail.com, Tel.: +91 9747630278

Available online at: www.ijcseonline.org

Abstract— We present an automatic occlusion removal methodology for occluded images. The occlusion here considered are the images which contain elements such as grid or fence, the reflection of objects through glass windows and raindrop. The appearance of any object in the space which blocks the complete view of another object or a scene considers as the occlusion. Because of occlusion, we lost the aesthetic beauty of the desired scene. To obtain a background scene without any discrepancies occlusion removal is essential. Occlusion may happen accidentally, and also there are some situations we cannot avoid occlusion. For example, taking photos or videos in a zoo, fence removal is impossible. If the fence obstruction removes from photos, the results became awesome. The Aim is to improve the accuracy of occlusion removal system using an optical flow method. The sequence of frames considers as input to the system. The system automatically detects occlusion. The decomposition of the background component and occlusion components are done using an optical flow method. Finally estimates desired background scene while removing the annoying occlusion. We show results of experiments in the various occluded situation while taking photos or videos.

Keywords— Occlusion, Optical flow, Detection, Decomposition.

I. Introduction

Now we are living in a world in which people love photography. They want to capture the photos of every occasion as wells as every moment in their life. They want to keep it as a memory in their life.

Nowadays computational photography techniques are available. New processing techniques are emerging to improve the quality of images and videos. But many photographic conditions are far from the most favorable results. For example, visual occlusions forcing us to take photos through the occluded region that we cannot avoid while we are capturing visuals. Occlusion is a problem in the field of photography.

The occlusion appeared in between the camera, and the desired scene wants to capture. Some real-world situations were forcing us to take photos through the occlusion. Significant occlusion includes some real-world objects, fence, reflection, and also raindrop causes occlusion during the raining season.

Usually taking photos through a fence, the fence, grids makes the disturbance in the real view of the image. Actually, during the shooting, the removal of the fence grid is an impossible task. Especially the protected areas like the zoo with a camera surveillance system. There the animals are in a cage or a particular protecting infrastructure. No other way to shoot through an enclosure.

Visual occlusions are often impossible to removeusing the camera techniques. Computational techniques not so much improved to remove an occlusion from the image with ease. A robust image processing computational approach is required to solve the occlusion problem. Through this study, we are focusing on the occlusion handling algorithms. Aim to implement more accurate automatic occlusion removal technique through which improves the quality of images and videos.

In this paper, we present an automatic occlusion removal algorithm that allows a user to take photos through an occlusion producing the occlusion free image as the output. Our algorithm requires the user input as a video or a sequence of image frames while moving the camera horizontally from one end to another. Occlusion detection and removal processing are fully automatic.

While considering the various occlusion situations like physical occlusion, semi or fully transparent occlusion, required a common accessible experimental setup. In case of reflection or object occlusion, the captured image contains a background scene with the overlying layer of occlusion. As a result image pixels have the different depth than the desired background image that we want to capture. Thus instead of taking a single picture, it is required to catch a sequence of images while moving the camera. Our algorithm considers the pixel intensity difference in image sequences and variations in optical flow motions in layers.

Our algorithm produces two components: an image of the background scene, and the occlusion.

The occlusion was handling from images in the past under different methods. Here we review related work in those areas. Various methods were proposed to handle occlusion. GPU accelerated method is the obstruction-free photographic technique [4]. Fence removal algorithm [5] and raindrop removal[6,7] algorithms are another important works.

Rest of the paper is organized as follows, Section II contains the related work of occlusion handling, Section III contains some measures of optical flow method, Section IV contains the architecture and essential steps of Automatic occlusion removal system, section V explains the optical flow methodology with flow chart, Section VI describes results and discussion of our method, Section VII contains the recommendation of performance evaluation and Section VIII concludes research work with future directions.

II. RELATED WORK

Here we review related work in those areas. Occlusion removal methods are related to the image and video processing techniques [1]. To remove an object from a video or image source, the user must mark the object region manually. The user marks some occluded region and fills using the information from the neighboring pixels. When the input and occlusion mark is given to the system, it identifies the occlusion pixels then immediately removes the occluded pixel values. After removing an occlusion, some holes appear in the entire image. The missing pixels are filled using the values of neighborhood pixels. Image and video in-painting techniques are focused on the object removal [2]. In this case, the system asks the user to specify the occlusion along with the input. This in-painting method is a manual method. The work focuses on light field occlusion removal techniques. Texture-based synthesis technique and in-painting techniques [3,12] improves the quality of the reconstructed image.

There were several works based on occlusion handling of images and videos. In such works, occlusion detection has a vital role. Occlusion detection mainly focuses on the method to fill unknown pixels of the image. Here occlusion detection [4] and removal methods applicable to light field images.

Several other papers proposed to detect and remove partial occlusions from images and also from videos. Several methods proposed to fence occlusion removal [5,6,7]. It involves the regular structure removal. The work [8] proposed to detect the snow and raindrops in images. It uses frequency space [9] for raindrop removal, measures of physical properties is the more significant factor.

The optical flow estimation technique used to determine whether a particular image region is occluded or not. The probabilistic formulation methodology [10] identifies occluded areas in the image. Estimated noise model used for this purpose. It generates the histogram of occluded pixel intensities. The level set method [11] along with a threshold to determine an occluded region in the image. Above mentioned two methods are trying to minimize the energy function by iteration. All these work based on the obstruction category and visual properties of the obstruction. The obstruction category involves reflection surface as imaging through a glass window, partial occlusions like a fence or the grid, and semitransparent occlusion like raindrops.

III. METHODOLOGY

Occlusion removal technique consists of two approaches. First one is the manual approach and the second automatic approach for occlusion removal. Manual approach: The input given to the system is a sequence of images includes five frames. The data formed by using a horizontal camera moving from one end to the other end. Along with the data input, five corresponding occlusion mask given to the system. Occlusion mask is created by the users specifying the object for removal by drawing the thick line over the occlusion region.

Figure 1.Hanoi_input_1*(First frame in the input sequence)*

Figure 1 shows the first frame in the Hanoi_input sequence out of 5 consecutive frames. The occluded region marked shows in Figure 2.

Figure 2.Hanoi_input_mask_1*(Occlusion Marked First frame)*

Figure 3.Hanoi_output_1*(First frame in the output sequence)*

In the resulting image Figure 3, the occluded regions remove from the input and get an occlusion-free image. The output image sequences stores in a median stack with label median_stack_1 to median_stack_5.

In our framework, there are three phases involved to achieve the occlusion removal along with the optical flow method: Occlusion Detection, Median Filtering, and Stack Propagation. The Architectural design of the Manual approach is shown in Figure 4.

The occlusion detection stage computes the motion field for each layer. Along with the motion field, our algorithm uses Edge Flow algorithm [4,14] to determine the flow between edges in each background and occlusion layer. Canny Edge detector [13] uses to detect edges and corners. From this, the transformation is calculated to align the results in a focal stack and takes an average. Calculate the difference in the intensity of each pixel withan average value. By it, the pixels are assigned either into the background layer or occlusion layer.

Median Filtering is the essential step in this occlusion removal method. After loading the input into the focal stack, Compute the median image from the focal stack and set as a reference frame. Lucas-Kanade Method is used to approximate the motion between the layers. Based on this focal stack and reference frame generate a transformation to the aligned images. Here in case of a fence or grid occlusion is the pure translation.

Median Filtering is used to compute edges. It works by moving through the image pixel by pixel, replacing each value with the median value of neighboring pixels. The pattern of neighbors is commonly called window which slides, pixel by pixel over the entire image to the pixel in the image. The median value is calculated by the following steps. Consider window which contains all pixel values. Initially sort all pixel values. Then replace the middle pixel in the window with the median pixel value.

Compute median image
med_filt = nanmedian(stack, number of frames)
There is a corresponding Confidence value in between zero and one. Confidence,

$$Ci = Di \quad * Pri \tag{1}$$

Where Di is the percentage of the known pixel over the total number of pixels calculated by using Equation(2).$Pri$is the similarity of the mean pixel Pm with Pi each pixel in the set of neighbours N. Pri calculate by Equation (3). U and V are pixel along x and y-direction and |UV| is set of unoccluded pixels.

$$Di = |UV| \div U + V - 1 \tag{2}$$

$$Pri = \sum_N \frac{\left(I(Pi) - I(Pm)\right)^2}{|UV| * (I_{\max})^2} / 2 \tag{3}$$

Focal stack propagation performs patch-based synthesis technique. It uses all in focus images to perform the synthesis. Then it checks the depth of each pixel in the image. Finally, the system generates a sequence of an occlusion-free image with more clarity.

IV. Architectural Design

Optical Flow Method: Optical flow is formed by the motion technique. Either the camera motion or object motion. Optical flow represents a particular pattern generated from the movement of object or camera between two consecutive frames during a small time interval. Optical flow is a vector. The vector formed by the change of position of an object in between two consecutive frames.The rate of change of position with time represents the displacement of the pixel, which is a vector with two dimensions, displacement in X coordinates and Y coordinates. In this case, the camera is moving horizontally from one end to another end then takes a sequence of image frames. Calculate the median image from the frame sequence then set as the reference frame.

Consider a 2D pixel intensity I with X coordinate, Y coordinate and time t in the 2D image input. Its X coordinates change its value by the small change in X-axis as dx and the small change in Y-axis dy after a small change in time dt.

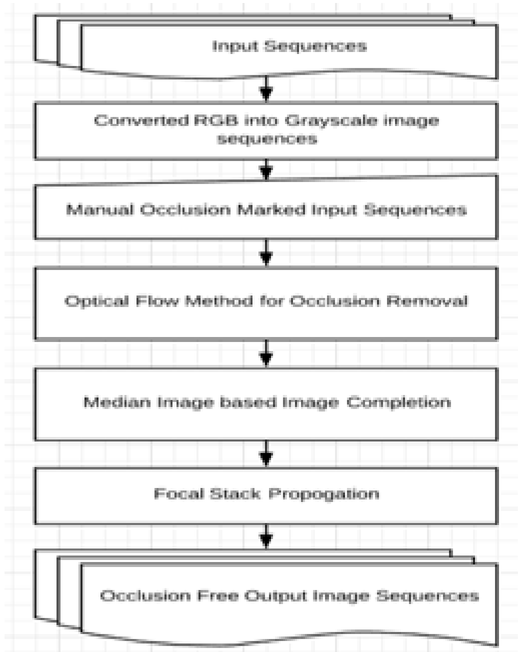

Figure 4. Manual Approach*(Architectural Design of Occlusion Removal)*

$$I(x,y,t) = I(x-dx, y-dy, t-dt) \quad (4)$$

Using Taylor series approximation on right hand side and divide both side by dt Equation(5) will obtain.

$$f_x u - f_y v - f_t = 0 \quad (5)$$

$$f_x = \frac{\partial f}{\partial x} \quad (6)$$

$$f_y = \frac{\partial f}{\partial y} \quad (7)$$

$$U = \frac{dx}{dt} \quad (8)$$

$$V = \frac{dy}{dt} \quad (9)$$

Where Equation (5) is known as Optical Flow Equation. In this Equation(5) f_x and f_y known as image gradients along X-axis and Y-axis. Similarly, f_t is the timegradient.But we cannot find value of (U,V) by system of linear equation. So Lucas-Kanademethod[13] applicable to get solution of the above equation[5].

Lucas-Kanade method: Lucas-Kanademethod[13] grab a current patch with neighboring pixels for example size of 3x3,6x6 or 9x9. All the neighborhood pixel points have the same motion. If we take a 3x3 patch, there must be nine neighboring points. Here nine equations with two unknown variables. In these situations, the system of the linear equation has no solution. To solve the problem, Equation(10) is useful.

$$\begin{bmatrix} U \\ V \end{bmatrix} = \begin{bmatrix} \sum_t f_{xt}^2 & \sum_t f_{xt} f_{yt} \\ \sum_t f_{xt} f_{yt} & \sum_t f_{yt}^2 \end{bmatrix}^{-1} \begin{bmatrix} -\sum_t f_x f_u \\ -\sum_t f_y f_u \end{bmatrix} \quad (10)$$

V. FLOW DIAGRAM

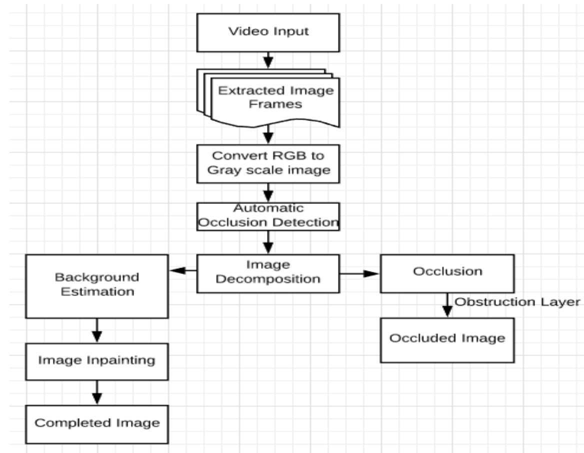

Figure 5. Automatic Occlusion Removal System *(Flow Diagram of Optical Flow Method)*

Automatic Occlusion Removal Approach: Consider the glass window reflections as an occlusion. When imaging through a window or capture a video of an outdoor scene through a glass window, the captured image represented as I. It consists of two layers. The desired background layer, I_B, and the unwanted obstruction layer I_O. The task is to decompose the input image, I into these two layers so that we can remove the obstruction from the input. If the system receives a video input, it automatically extracts sequence video frames then turns the color image into grayscale. After that, extract the reference frame in such a way that a frame in the middle to have a shorter motion vector is more precise. Then perform the remaining steps as shown in Figure 5. The occlusion detection, Image Decomposition steps are performed by the following algorithm. Image In-painting and image completion steps are used to fill the unknown regions of the image. Median image-based technique is used to perform in-painting, and patch-based texture synthesis is used to perform image completion.

Algorithm:

Data: Video or Image Sequences $\{I_t\}$ at time t.

The Initial guess of I_B, I_O, Matrix of transformation A.

Result: I_O, I_B, Matrix of transformation A, U, V

Step1: Set median image as the reference frame.

Step2: Canny edge detection on the reference frame.

Step 3: initialize the background,

I_Background(x,y)=255;

Step4: Extract every frame and apply edge detection.

Step5: Determine Optical flow by Lucas Kanade

5.1: Find corners

5.2: Reduce the size of the image.
5.3: Discard the corners near the margin.
5.4: Lucas Kanade Method
5.4.1: Ix_m=conv2(I1_edge,[-1,1;-1,1]); Partial on x
5.4.2: Iy_m=conv2(I1_edge,[-1,-1;1,1]); Partial on y
5.4.3: It_m=conv2(I1_edge,one(2),'valid')+ conv2(I2_edge,- one(2),'valid');
5.4.4: u=zeros(length(c),1)
5.4.5: v=zeros(length(c),1)
5.5: Calculate Affine transform A=[Ix,Iy]// Result A obtained.
Step 6: Visualize Optical flow vectors, I2_edge
Step 7: Find Minimum Motion vector using Euclidean distance,
Sqrt((transition(i,1))2+(transition(i,2))2);
Vmin=[transition(row,1)transition(row,2)];
Vobstruction=[transition(row_max,1)transition(row_max,2)];
//Occlusion obtained
Step 8: Extract Background,
I_background(a,b)=min(I_background(a,b), I2_translate(a,b)); I_obstruction(a,b)=I1(a,b)-I_background(a,b);
ifI_obstruction(a,b)< Threshold
I_obstruction(a,b)=0; // I_B , Background Layer without occlusion,
else
I_obstruction(a,b)=1; I_O, Occlusion Layer
Step 9: End

VI. RESULTS AND DISCUSSIONS

We tested our algorithm in various situations with occlusion appears. It worked successfully and resulted obtained. The algorithm generates a clean separation of the occlusion layer and background layers. Tested our algorithm with the images of occlusions as fence, reflection, raindrops and moving object appeared as occlusion. In all cases, the algorithm produces the proper reconstruction of background scene with the occluded content removed.

Quantitative Evaluation: To evaluate results quantitatively by calculating the normalized cross-correlation (NCC) of recovered decomposition with ground truth decomposition. The following equation calculates the NCC of two images. Let original_image is the ground truth image,output_image is the result of our algorithm. Normalized Cross-correlation, NCC = sum(sum (original_image*output_image)) ÷ sum(sum (original_image*original_image)).

The NCC of our recovered occlusion free background images with the ground truth backgrounds were Manual approach occlusion removal Hanoi_input, NCC=1.005269, and Automatic occlusion removal using optical flow method for reflection removal from a video input, background scene NCC=0.9975631.

Table 1. NCC Value of Occlusion Removal Methods

Occlusion Removal Method	**NCC Value for Recovered Background**	
	Fence Removal	***Reflection Removal***
[Li and Brown 2013]	0.9271	0.7906
[Guo et al.2014]	0.9682	0.7701
[Xue, Liu and W. T Freeman.2015]	0.9738	0.8985
Our Method	1.005269	0.9975631

Table 2. Optical flow method Fence Occlusion NCC Value

Fence Occlusion NCC with Ground truth image		
Input Sequence	***Algorithm Result***	***NCC Value***
Hanoi_input_1.png	Recovered Background_1	0.9270291
Hanoi_input_2.png	Recovered Background_2	0.9494236
Hanoi_input_3.png	Recovered Background_3	0.9526250
Hanoi_input_4.png	Recovered Background_4	0.9596487
Hanoi_input_5.png	Recovered Background_5	1.005269

Table 3. Optical flow method Reflection NCC Value

Reflection Removal NCC with Ground truth image			
Estimated Background			***Estimated Occlusion***
Input Video	***Result***	***NCC Value***	***NCC Value***
Frame 1	Bg_1	0.8998939	0.3846623
Frame 2	Bg_2	0.9432619	0.4099005
Frame 3	Bg_3	0.9555014	0.5081293
Frame 4	Bg_4	0.9930570	0.6731001
Frame 5	Bg_5	0.9975631	0.7521180

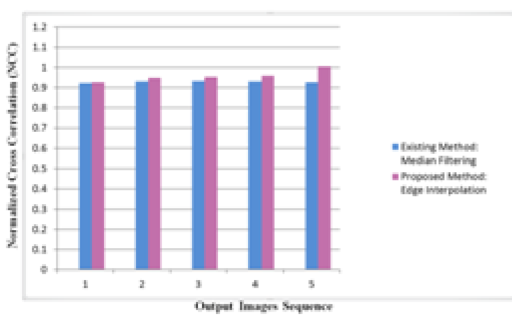

Figure 6. Comparison of Performance Evaluation Measure NCC *(Fence Occlusion)*

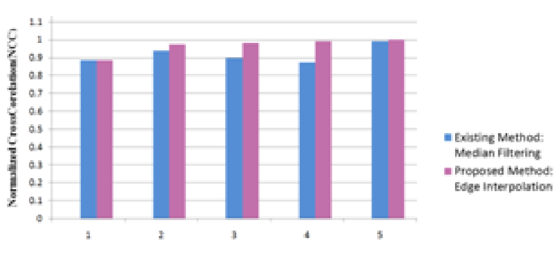

Figure 7. Comparison of Performance Evaluation Measure NCC*(Reflection Occlusion)*

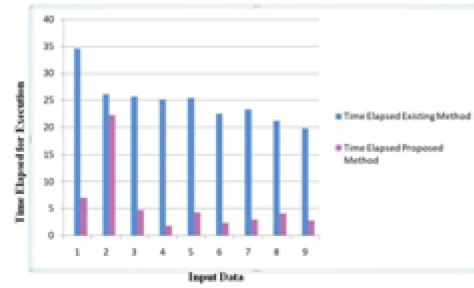

Figure 8. Comparison of Performance Evaluation Time*(Rain Drops Occlusion)*

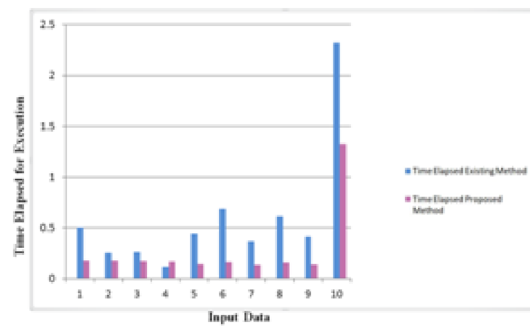

Figure 9. Comparison of Performance Evaluation Measure Time*(Reflection Occlusion)*

VII. CONCLUSION AND FUTURE SCOPE

In this paper, we have demonstrated occlusion removal using an optical flow method. Our algorithm increases the quality of photos, and we obtain a background scene with more quality. Our algorithm performs with the sequence of images. Handling occlusion removal with the single photograph is a challenging task. Using image sequences our algorithm produces the better result.

REFERENCES

[1] Criminisi, Antonio, Patrik Perez, and Kentaro Toyama, "*Region filling and object removal by exemplar-based image inpainting*", IEEE Transaction on image processing 13.9, pp.1200-1212, 2004

[2] Newson, A., Almansa, A., Fradet, M., Gousseau, Y., Perez, P., et al. " *Video inpainting of complex scenes.*", Journal on Imaging Sciences, Society for Industrial and Applied Mathematics, 2014

[3] Kao, Shannon, "*Light Field Occlusion Removal*", Sanford University, 2005

[4] TianfanXue, Michael Rubinstein, Ce Liu, William T Freeman, " *Computational Approach for Obstruction -Free Photography*", ACM Transactions on Graphics, Vol. 34, No. 4, Article 79, 2015.

[5] Park, Minwooi et al. "*Image de-fencing*", Asian Conference on Computer Vision. Springer, Berlin, Heidelberg, 2010

[6] Liuy, Yanxi, et al. "*Image de-fencing*", Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on IEEE, 2008

[7] Jonna, Sankaraganesh, et al. "*A multimodal approach for image de-fencing and depth inpainting*", Advances in Pattern Recognition (ICAPR), 2015 Eighth International Conference on. IEEE, 2015

[8] Yamashita, Atsushi, Akiyoshi Matsui, and Toru Kaneko. "*Fence Removal from multi-focus images.*", Pattern Recognition (ICPR), 2010 20th International Conference on IEEE, 2010

[9] Garg, Kshitiz, and Shree K. Nayar., "*Detection and removal of rain from videos*", Computer Vision and Pattern Recognition, 2004. CVPR 2004, Proceedings of 2004, IEEE Computer Society Conference on.Vol.1. IEEE, 2004

[10] Strecha, Christoph, RikFransens, and Luc Van Gool., "*Wide-baseline stereo from multiple views: a probabilistic account.*", Computer Vision and Pattern Recognition, 2004. CVPR 2004, Proceedings of 2004, IEEE Computer Society Conference on.Vol. 1. IEEE, 2004

[11] Luan, Xiao, et al., "*Extracting sparse error of robust PCA for face recognition in the presence of varying illumination and occlusion.*", Pattern Recognition, 47.2 PP. 495-508. 2004

[12] Ashraf Siddique and Seungkyu Lee., "*Video Inpainting for Arbitrary Foreground Object Removal*", IEEE Winter Conference on Applications of Computer Vision, 2018

[13] VanshajSikri., "*Proposition and Comprehensive Efficiency Evaluation of a Foreground Detection algorithm based on Optical Flow and Canny Edge Detection for Video Surveillance Systems* ", IEEE WiSPNET conference. 2016

[14] T. Brox, A. Bruhn, N. Papenberg, and J. Weickert. "*High accuracy optical flow estimation based on a theory for warping*" European Conference on Computer Vision (ECCV), 2004

[15] OpenCV Open source Computer Vision_optical Flow(https://docs.opencv.org/3.3.1/pages.html)

Authors Profile

Ms. M A Aneesha pursed Bachelor of Technology from the University of Calicut, 2014 she is currently pursuing M.Tech Computer Science and Engineering, Department of Computer science and Engineering, Government Engineering College, Thrissur, under University of APJ Abdul Kalam Kerala Technological University, Thiruvananthapuram, Kerala 2016-2018. Area of interest is Image Processing and Computer Vision.

Mrs. K J Helen, pursed B. Tech (Computer Engineering), CUSAT, M.Tech (Computer Science), NIT Thiruchirapally. She is currently working as Associate Professor in the Department of Computer Science and Engg. Govt. Engineering College, Thrissur under University of APJ Abdul Kalam Technological University, Thiruvananthapuram, Kerala. She has 18 years of teaching experience and areas of interest are Image Processing and Computer Networks.

