

Principles of Computer Vision for AI

Aiden Williams

372001L

aiden.williams.19@um.edu.mt

Logan Formosa

434901L

logan.formosa.19@um.edu.mt

Part 1

Stage 1

In the first stage of the assignment, we were tasked with retrieving an object from a scene and removing its contents from the original and displaying it in other scenes. To approach this problem a dataset of such objects was first retrieved containing different scenes with multiple objects of varying types and sizes. The COTS dataset [1] was used in which such scenes were provided against a green backdrop as well as other complex scenes containing a changing lifelike background instead of a backdrop.

For each object present in a scene within the dataset, a mask for that specific was provided. These masks highlight the region occupied by the object of interest within the image. Object Masks are a binary image, in which the white region denote that the space is occupied by the concerned object. With a combination of opencv's functionalities and these object masks on the test images we were able to extract specified objects from these images and use these extracted objects for other operations.

ExtractObject()

The ExtractObject() function takes two parameters, the original scene containing the object that needs to be extracted, and the mask of the required object to be extracted. The function performs a bitwise and operation on both the original scene S2 and the supplied mask to return the common region between the two images, the result being a blank image except the region, which is occupied by the object, in which it is filled by the contents of that object instead. The below figures illustrate an example of this function and its resultant extraction.



Fig 1: An example scene, S2, containing three objects.



Fig 2: The Mask of the third statue within the image, this is the object we wish to

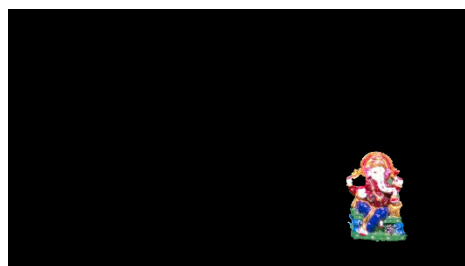


Fig 1: The resultant extracted object.

ApplyFilter()

Next, some convolutional filters are applied on the extracted object. This is done to aid in making the extracted object's transition into other images seem less fabricated and artificial since with this method one can visually see sharp jagged edges at the outline of the object when it is inserted into another scene S1 if not subjected to a convolutional filter that helps smoothen the image first.

Three convolutional filters were implemented using pre-defined kernels that iterate over the image passed to the function as a parameter. These filters were:

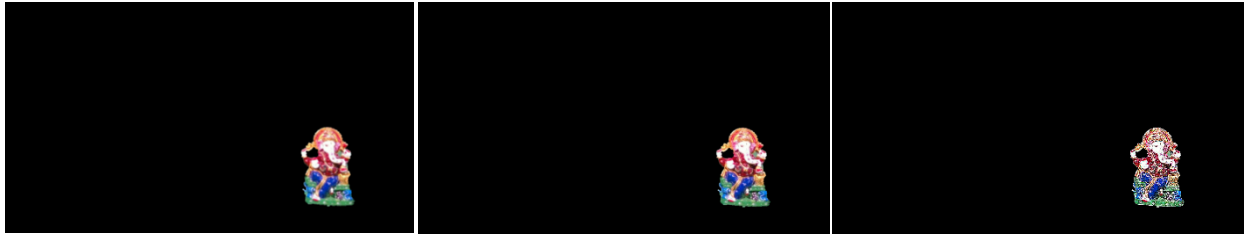


Fig 2: From left to right, 5x5 Averaging, Gaussian Blur, Kernel Sharpening

ObjectBlending()

Next, the extracted object was to be placed in another scene, S1. To achieve this the image containing the extracted object was passed as an argument as well as the new scene S1. Then the image containing the extracted object is iterated through, and at any point in which the pixel value is not black, or 0, the value is placed onto S1 at the same co-ordinates, provided that S1 is of the same dimensions as the original S2 from which the object was extracted will ensure that the extracted object is placed in the new scene with the same position.



Fig 5: A scene S1 with a single object.

CompareResult()



Fig 6: A Blended image, adding the extracted object onto S1.

Finally, an error metric was conducted on an original image with actual objects and our blending result with the same objects being artificially placed in the image. Two error metrics were used, these are the Sum of Squared Distances (SSD) and the Mean of Squared Error (MSE) by comparing the difference in pixel values between the two images. The following results were achieved when blending the same object onto a scene with different convolutional filter being applied before blending.

Error Metric	No Filter	5x5 Averaging	Gaussian Blurring	Kernel Sharpening
SSD	161627794	1690953909	167956935	170218575
MSE	58.5491	61.1600	60.74831	61.5663

The high values found in SSD are due to the lack of shadows when comparing the original object and the blended object. Since the extraction only takes the object itself, it is blended without any regards to the current lighting and so its shadows are missing, these missing shadows then add up to a high squared distance error due to the difference in contrasting light and dark region caused by a lack of shadow. MSE provides smaller error values, since these are being normalised by the size of the image, considering what percentage of the region is in fact at such a large error, so the large error values caused by the small regions of missing shadows do not have such a large effect in this metric.

Stage 2

In this stage of the assignment two functions were developed: `removeGreen` and `changeBackground`. The two functions operate similarly.

The process starts by converting the passed image, like Fig 7, to HSV colouring and then getting a mask of the green colours.

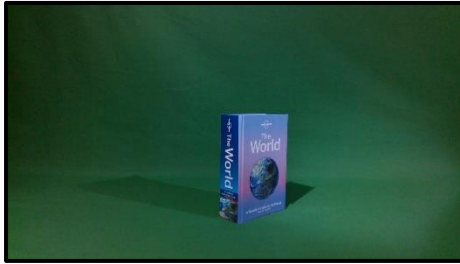


Fig 7: COTS Dataset Book Image

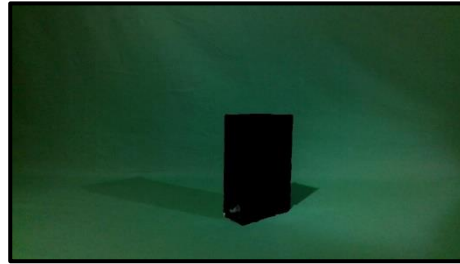


Fig 8: Green Mask of the input Image

When this mask is removed from the image the result is the green parts of the image as in Fig 8.

Figs 9-12 shows the grayscaling of the result, thresholding, not operation and closing. Closing is done so that green within the objects in front of the greenscreen is included in this newly generated mask.

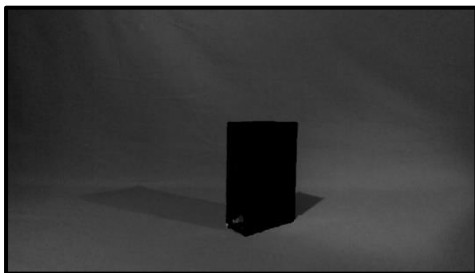


Fig 9: Grayscale of Fig 8

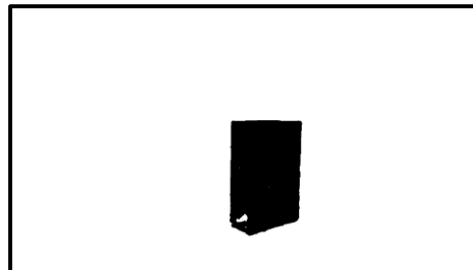


Fig 10: Binary Threshold of Fig 9



Fig 11: Not Operation on Fig 10



Fig 12: Final Mask after Closing Morph

The final step is that every pixel in the passed image is checked. If the corresponding image from the new mask is not white then this pixel is set to black. The result is Fig 13.

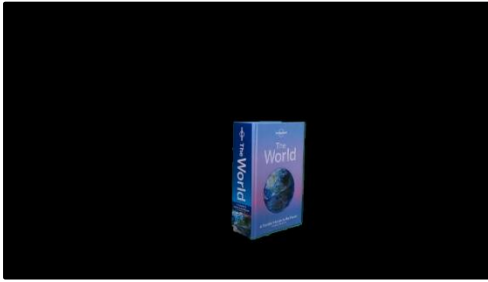


Fig 13: Green Background Removed from input Image

In the case of `changeBackground` the final step differentiates by painting the object onto a new background instead of a black background like in Fig 14. For backgrounds larger than the passed image, the image will retain its original coordinates this can be seen in Fig 15.



Fig 14: Changed Background to 1280x720p



Fig 15: Changed Background to 1920x1080p

Figs 15-16 show different source images have their background changed to different 1280x720p backgrounds. These figures also highlight an advantage and disadvantage of using the closing morph



for the mask. In fig 16 the Ganesha figure features the god sitting on a green seat, because of the closing morph parts of this get included in the final image. The green background between the god's hands and head, however, is also included, and some of the seat is still left out. Similarly in fig 17, the green background can still be seen between the Buddha's hand and torso.

Fig 16: Changed Background to 1280x720p using a different COTS image



Fig 17: Changed Background to 1280x720p using a different COTS image

Part 2

Task A

In this task we were required to implement the evaluation code in [2] by using off-the-shelf OpenCV inpainting methods. The TELEA and NS inpainting techniques were used, while the same S2, Mask and S1 images as used in [2] were used here. In Table 1, the image set used and the results of the inpainting is shown, while in Table 2, the SSD and MSE metrics are displayed.



























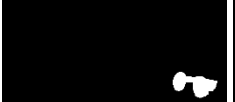



Label	S2	Mask	S1	TELEA	NS
Statues					
Shooter Glasses					
Academic Books					
Footwear					
Mugs					
Tech					

Table 1: Inpainting Visual Results and Input Images

In Table 1 The difference can be seen in S1 and the inpainted results as the inpainting was not as good as the real image. By using the SSD and MSE metrics we can see both TELEA and NS perform better in lower lighting settings, as can be seen when comparing the 'Statues' and 'Shooter Glasses' results.

Label	SSD (Telea)	SSD (NS)	MSE (TELEA)	MSE (NS)
Statues	487461624.0	496995348.0	42.20	42.31
Shooter Glasses	83845946.0	84640814.0	13.17	13.19
Academic Books	372850560.0	379575607.0	23.02	22.73
Footwear	87419718.0	96006167.0	17.64	17.57
Mugs	78877863.0	83528925.0	14.98	15.16
Tech	132791205.0	146494060.0	14.79	15.01

Table 2: Inpainting Metric Results

Task B

In this task we used the same code and process from Task A on a new dataset, featuring dynamic backgrounds. 3 Sets of objects are used, each having a W (Wind) and NW (No Wind) version. Also, mask extraction was performed as masks were not provided for each individual item in separate files but rather in one file with different colours. The same presentation is used here with Table 3 representing the visual inputs and outputs and Table 4, the evaluation metric results.


























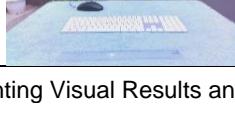
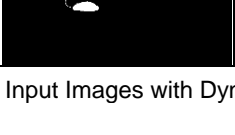
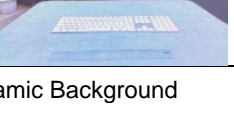


Label	S2	Mask	S1	TELEA	NS
Books A NW					
Books A W					
Bottles A NW					
Bottles A W					
Electronic s A NW					
Electronic s A W					

Table 3: Impainting Visual Results and Input Images with Dynamic Background

Here Table 4 shows that the previous assumption that related image darkness to lower MSE score wrong. As can be seen in the 'Electronics A W' results, it is one of the darkest images present but also has the highest MSE score. At the same time 'Electronics A NW' has the lowest MSE score. Here the high and low scores could be attributed to the fact that 'NW' images have less noise appearing on the background than 'W' labelled images.

Label	SSD (Telea)	SSD (NS)	MSE (TELEA)	MSE (NS)
Books A NW	872725274.0	952867397.0	24.76	24.33
Books A W	967066154.0	1079077087.0	25.32	25.45
Bottles A NW	246319032.0	257172040.0	22.48	22.69
Bottles A W	1070677610.0	1108628015.0	34.39	34.37
Electronics A NW	415069437.0	469780795.0	15.22	15.32
Electronics A W	389166321.0	408671591.0	49.26	49.30

Table 4: Dynamic Background Impainting Metric Results

Background Changes Visualisation

The complex backgrounds presented in the dataset were not static, an element of Wind was present in instances which altered the configuration of the surrounding leaves and affected the background. Other elements such as different lighting on reflective surfaces could also be discerned as a visual change from either the background or the foreground.

A method used to visualise these changes between the same backgrounds is called Background Subtraction. In this method the difference between values of two images is constructed by looking at the individual pixel values. Images are first converted to grayscale to aid in processing, then an initial image is passed which will serve as the focal point on which changes in the background are calculated against. A second image is then passed and a mask is constructed by seeing which regions between both images feature a discrepancy in values, those which do not match will be featured in the binary mask as a white value.

This method has several applications that go beyond visualising changes in background elements, when paired with a continuous video stream and an initial frame of an empty scene with no objects obscuring the background, background subtraction can be used to detect objects in real-time and separate the foreground from the background as well.



Fig 18: Before Image



Fig 19: After Image

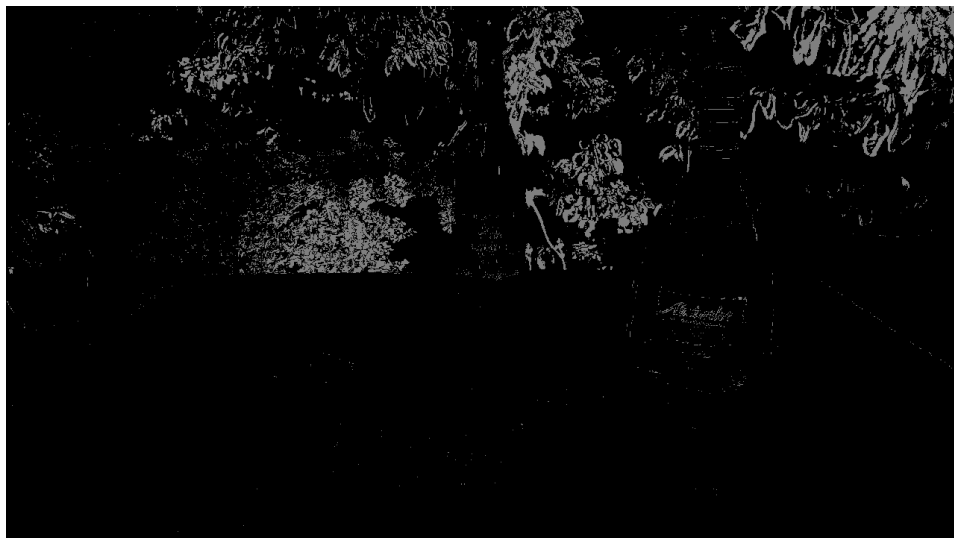


Fig 20: Difference Between Before and After Images

From the above figures we can see that the biggest change in the background occurred in the leaves on the back, since the dataset informed us that the complex background features an element of wind difference between certain sets of images, the resultant visualisation is as expected. The silhouette of the objects is easily identified as these objects remain static in both images and therefore a clear

outline of the changing-background and the objects which remained the same, clearly separating the foreground and the background, can be observed.

References

- [1] D. Seychell, C. J. Debono, M. Bugeja, J. Borg and M. Sacco, "COTS: A Multipurpose RGB-D Dataset for Saliency and Image Manipulation Applications," *IEEE Access*, vol. 9, pp. 21481-21497, 2021.
- [2] D. Seychell and C. J. Debono, "An Approach for Objective Quality Assessment of Image Inpainting Results," *2020 IEEE 20th Mediterranean Electrotechnical Conference (MELECON)*, pp. 226-231, 2020.