Using weighted median filtering to eliminate moving objects from a series of photographs

Caroline Bridge
CIS*4720: Image Processing and Vision
University of Guelph
Guelph, ON, Canada
bridge@uoguelph.ca

Kevin Sullivan
CIS*4720: Image Processing and Vision
University of Guelph
Guelph, ON, Canada
ksulli06@uoguelph.ca

***Abstract*—Eliminating objects from photos can be done in numerous ways. The following paper will analyze an implementation of one such method and its effectiveness of removing a moving object from a sequence of images.**

***Keywords*—median filtering, unwanted objects, removal, alignment, algorithm, analysis, image processing**

I. INTRODUCTION

Eliminating objects from photos is a problem that can be practically applied to everyday life when attempting to capture images that are often populated with people or other undesirable moving objects. Given a series of images, it is possible to eliminate the objects moving across the images using any number of implementations of weighted median filtering. The following paper investigates the effectiveness and implementation of weighted median filtering in its elimination of moving objects from images.

II. ALGORITHM DISCUSSION

A. *Terminology*

For the purposes of this paper, certain terms are used to describe the quality of the images and image sequences discussed. They are described in Table I.

TABLE I.TERMINOLOGY

Term	**Description**
Frame	A single image in a series of images used to create the result
Sequence	A series of images used to create the resulting image
Standard Sequence	6 Frames
Long Sequence	12 Frames
Ghost Image	The artifact that results from a series of images with camera shake that has been processed with median filtering
Overlap artifact	The artifact that results from a series of images with severe subject overlap that has been processed with median filtering
Degree of Camera Shake	
None	A tripod has been used to capture this sequence. There is no camera shake whatsoever.
Very Little	A steady hand or other propping mechanism has been used to capture this sequence.
Some	An unsteady hand has been used to capture this sequence.
Severe	The camera shake in this sequence is intentionally severe and is more camera shake than a standard set of hands might capture.
Degree of Overlap	
None	The subject in the sequence does not overlap over any frames
Some	The subject in the sequence overlaps over one or two frames
Severe	The subject in the sequence overlaps over most or all frames

B. *Median filtering*

There are three kinds of basic statistical filtering to consider when working with image processing. Mean, median, and mode filtering are all used to evaluate pixels in a neighbourhood and apply different properties onto these pixels depending on the statistic. In the case of attempting to filter out objects from images, it was discovered that median filtering yielded the best result. Mean filtering would take the average of the neighbourhood and replace the pixel with an artificially calculated value which may or may not actually exist in the image. Mean filtering output will therefore fluctuate too much across images and is omitted from algorithm design. Mode filtering is better suited than mean as it will only use existing pixel values, but suffers from probability fluctuation. Mode filtering is using the pixel value which is most commonly found in the neighbourhood. This filtering method appears to be well suited for the algorithm, but returns more fluctuation in results. There is less certainty in selecting the correct pixel values using the mode because of cases having multiple correct values and all values being different. These cases deem the mode as being an unsafe implementation in the algorithm. Median filtering is more closely related to the mode. The median, however, will select the middle pixel value in a sorted array. This guarantees an existing and safer pixel value to be used. In a lot of cases, the median can pick the same value as the mode, but statistically has a higher chance of choosing a value more realistic.

C. *Alignment of images*

Image alignment is an integral step in creating the algorithm. Images that are taken with a handheld camera will inevitably suffer from camera shake. This creates the problem of ghost images when applying median filtering. Camera shake means that pixel values will be shifted across images and weaken the median filtering from producing a viable result.

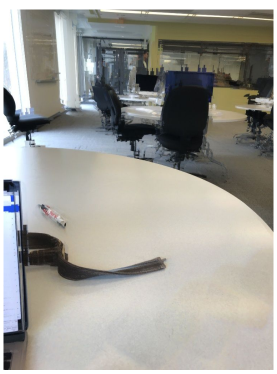

Fig. 1. An image without alignment applied after being processed through median filtering.

When alignment is not applied, the median filter will produce an image like the one seen in Figure 1. The image suffers from multiple incorrect pixel values and ultimately creates an unrealistic photograph.

```
image_array = getImages()

base_file = image_array[0]

aligned_array = []

for curr in image_array {

    aligned = warpAffine(curr,  base_file)

    aligned_array.append(aligned)

}
```

In order to get the most accurate array of aligned images, the affine transformation is applied to each image based on the first image. Aligning each image with its predecessor returned less optimal results as slight variations would increase gaps in actual image placement as the loop iterates.

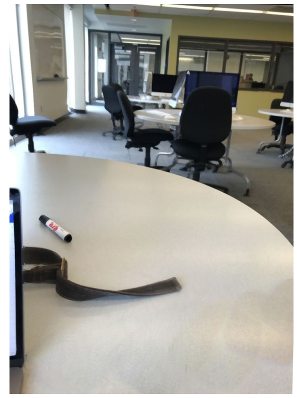

Fig. 2. An image with alignment applied after being processed through aligned-median filtering.

Instead, having a base image that each image in the array can consistently compare with creates a much better result. The result from the optimized algorithm can be seen in Figure 2.

D. *Cropping out alignment edges*

Translating images through alignment will pull images outside of their recorded pixel values. This creates a series of vertical and horizontal bars across the image. When median filter is applied, these black bars will remain in the resulting image, creating an unappealing output. These boxes are relatively easy to remove through the process of cropping. The black bars are mostly pure black with some potential fluctuations caused by the median filtering. This makes it easy to detect contours in the image with a reasonably low threshold range (10-255). Once these contours are found around the image, a bounding rectangle is used to compute the range (in x and y values) of the bars.

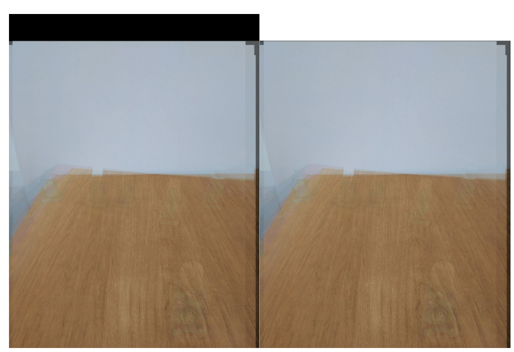

Fig. 3. (Left) An image processed through aligned-median filtering with residual borders versus (right) an image that has had the borders cropped.

The image is then cropped according to these values and successfully eliminates the black translation bars as shown in Figure 3.

E. *Colour correction*

To improve the overall quality of the algorithm's output, colour correction was employed. Using the OpenCV library available in Python, improving the quality of this image aspect was fairly trivial. The normalize() function takes a given image and normalizes the values given a threshold of 50 to 205. This eliminates areas that are too dark or too light and sets safer values for the next step. The image now iterates through each pixel to increase the α value by 1.8 and decease the β value by 50. If normalization was not applied, areas that may be too bright will get even brighter and could even cause pixel loss in the image. Normalization helps to ensure safer bounds for high and low intensity pixels.

$$g(x) = \alpha f(x) + \beta \tag{1}$$

In this case, f(x) represents the current image, the alpha value represents the contrast, and the beta value represents the brightness in (1). Decreasing the beta value will produce a brighter image while increasing the alpha value will add more contrast. Images are generally more appealing with these traits which is why they were included in the post-processing of the algorithm.

Figure 4 demonstrates the difference between the final result of an image that has been colour corrected versus one that has not. Overall, the image looks improved, and if the given algorithm were applied to a commercial application, the resulting processed image would be more satisfactory to a consumer.

Fig. 4. (Left) a processed donald_level1 without colour correction versus (right) donald_level1 with colour correction. Subjectively, the output has improved in overall quality in terms of visual appeal.

F. *Overlap of unwanted objects across the frames*

Testing showed that overlap of the object across multiple frames decreased the effectiveness of the algorithm, and should be avoided. Re-inserting a processed image back into the sequence does not improve the results of a processed sequence of overlapping images; the result is largely the same as if the image was not re-introduced (figure 5).Image overlap artifacts can be largely improved upon by taking photo sequences with a larger number of images in the sequence. Moreover, removing the offending images also improves the algorithm's performance, though this manual process can be tedious for a user. Note that automating this process is discussed in Section G.

Fig. 5. (Left) an aligned-median filtered processed image with the result image re-introduced and processed again versus (right) the same image processed without re-introducing the resulting image.

G. *Improving the algorithm*

Some improvements to the algorithm were planned, but the implementation of said improvements was out of the scope of this project. The first of these improvements was to apply deep learning to the established median filtering algorithm. Deep learning can be applied after our proposed algorithm to improve the performance of the algorithm further by removing artifacts left over from overlap and lack of data. Bayar and Stamm explain that a deep learning algorithm could be taught to recognize these tell-tale artifacts (transparent, repeating, etc.) would be able to remove them from the image and fill in the resulting gaps with the information that lies "behind" the artifacts (considering they are transparent and the actual information is there) [5]. This layered approach of identifying the artifact as what needs to be removed, removing it, and filling in the gaps, is hallmark of deep learning algorithms.

The second of these improvements, to achieve a similar result to the first, is an algorithm that automatically removes overlapping images, as discussed in Section F. This proposed improvement would detect areas between images that overlap by masking the object(s) that are not stagnant across the series of images and comparing them to the other frames in the sequence. Frames with overlapping subjects would be removed and the new reduced sequence would be put through the median filtering algorithm.There are a couple of caveats with this method, namely that reducing the number of frames available given a small dataset would cause the resulting image to have a "ghosting" effect caused by not having enough data for the background image. On 6-frame sets with heavy overlap however, this would, in theory, work well with the other aspects of the algorithm discussed in this paper.

III. EXPERIMENTAL RESULTS

A. *Summary of test data and discussion of testing metric*

Testing for expected output requires the use of a ground truth image. Once a result has been created from the median filtering, a difference calculation can be made on both images. The difference will show everything between the images that is not the same. Next, a binary mask is taken to ensure that the black pixels are true black and everything that is different between the images is white. From here, quantifying the data is simply summing the total black pixels and dividing them by the total pixels in the image. When multiplied by 100, this will give a score of the accuracy that the median filtered image has to the ground truth. This quantification proves to be very accurate when comparing its results with the human eye and is used in this report to accurately demonstrate the variance that certain factors cause.

```
ground_truth = readImage()

filtered = readImage()

diff = absDiff(filtered, ground_truth)

thresh = threshold(diff)

black_pixels = sum(thresh == 0)

total_pixels = diff.width * diff.height

score = black_pixels / total_pixels * 100
```

The following table details the testing data used and the test cases performed for the paper's discussion on algorithm results. Note that in the following table, "regular" refers to the median filtering algorithm's performance with 6 frames and no alignment, "many frames" refers to the median filtering algorithm's performance with 12 frames and no alignment, and "aligned" refers to the final proposed algorithm for eliminating moving objects in a series of images.

TABLE II. EXPERIMENTAL RESULTS

SEQU-ENCE	**Degree of camera shake**	**Degree of overlap**	**Number of frames**	**Test Case**	**Perfor-mance**
donald_ level1	none	none	6	Regular (no align)	99.99%
sanitizer	severe	none	6	Regular	82.50%
tim6	severe	severe	6	Regular	47.02%
donald_ level2	none	severe	6	Regular	98.89%
wallet	some	some	6	Regular	66.95%
shaky_ desk	severe	some	6	Regular	64.11%
tim12	severe	severe	12	Many frames	44.60%
donald_ level3	none	severe	12	Many frames	99.99%
shaky_ desk	severe	none	12	Many frames	64.87%
tim6	severe	severe	6	Aligned - final algo	87.21%
sanitizer	severe	none	6	Aligned	92.53%
wallet	some	some	6	Algined	81.07%
tim12	severe	severe	12	Aligned	78.91%

B. *Performance of regular median filtering*

Regular median filtering is defined as the proposed median filtering algorithm without image alignment or other post-processing, and considering a standard sequence of frames (6) for a given test case. It was hypothesized that the algorithm would perform fairly well on the images, given the nature of applying median filtering, however the results indicate that additional processing is required to produce perfect output.

The regular median filtering algorithm is first tested with a sequence of images with no overlap of the subject across the frames, and no camera shake, as seen in figure 6. The regular median filtering algorithm performs with a success of 99.99%, which was expected given that the sequence does not have any "flaws". As median filtering takes the middle value of a sorted list of pixels for a given area of the image, the pixels, unhidden by overlap or camera shake, turn out to be the same as the ground truth image.

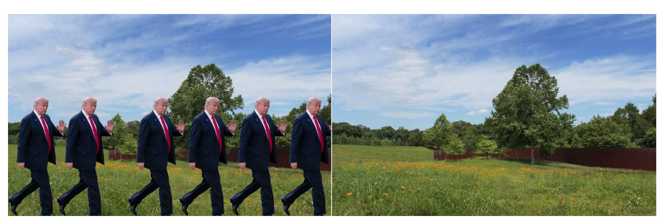

Fig. 6. (Left) the donald_level1 sequence superimposed together versus the (right) resulting image after processing through the regular median filter algorithm.

When overlap is introduced to the sequence, shown in figure 7, the performance of the regular median filtering algorithm is diminished to 98.89%. The data "behind" the moving subject is now obscured given the overlap, and the algorithm performed worse. The resulting image has overlap artifacts that blight the resulting image, but still removes the majority of the moving subject from the image. This is the first indication of the limitations of a regular median filter applied to a sequence of images to remove moving objects. It was hypothesized that increasing the number of frames in the sequence would produce a better result given that though the overlap still exists in areas of the image, the algorithm would have more data to draw upon to conclude a median for the area.

Fig. 7. The result of donald_level2 put through the regular median filter algorithm, which introduces overlap of the object across the background image. The result features some overlap artifacts.

The regular median filtering algorithm is next tested with an image with no overlap and severe camera shake, which is shown in figure 8. A similar, but distinct effect is leftover by processing images in this category with the regular median filtering algorithm. This ghosting artifact is the result of the camera shake of the image, where the difference between the "background" of the image leaves artifacting where there would otherwise be nothing, because the algorithm does not have an idea of what the "background" actually looks like. The regular median filtering algorithm performs with a success of 82.50%, which indicates that more needs to be done to eliminate the camera shake of the frames in the image before being put through the median filtering algorithm. Aligning the images is hypothesized to increase the success of the algorithm and is discussed later in this paper.

Fig. 8. (Left) the sanitizer sequence superimposed together versus the (right) result of sanitizer put through the regular median filter algorithm, which features severe camera shake. The resulting image features ghost artifacts.

Figure 9 demonstrates that, when processed through the regular median filtering algorithm, a severe instance of both camera shake and overlap worsens the presence of both artifacts in the image. The success value of 47.02% shows that fixing both of these will achieve a more desirable resulting image.

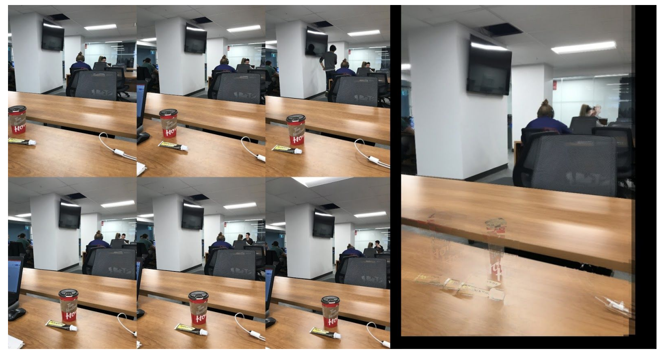

Fig. 9. (Left) the tim6 sequence superimposed together versus the (right) result of tim6 put through the regular median filter algorithm, which features severe camera shake and severe overlap. The resulting image features both overlap and ghost artifacts.

A similar, but more realistic test case can be seen in figure 10, where camera shake and overlap are only slightly present. This case represents a "typical use case" of the algorithm, and performs with a success value of 66.95% which shows the linearity of the increase/decrease of both artifacts. That is, that an decrease in alignment and a increase in overlap results in a poorer result as the severity of the flaws grows.

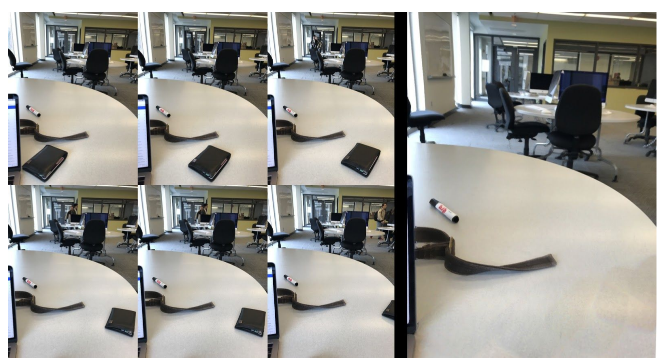

Fig. 10. (Left) the wallet sequence superimposed together versus the (right) result of wallet put through the regular median filter algorithm, which features some camera shake and some overlap. The resulting image features both overlap and ghost artifacts.

C. *Testing the impact of number of frames*

From the results of processing images with a regular median filtering algorithm, it was noted that overlap of the subject across one or more frames left artifacting in the resulting image. It was hypothesized that this was caused by the sequence not having enough data of the "background" image with respect to the moving object to successfully produce a satisfactory result. To test this theory, more frames were introduced to the sequence. For this section, all test cases have 12 frames.

With severe overlap and an increased number of frames from 6 to 12, the success value of the test case improves from 98.89% in the previous section to 99.99%. The resulting image is depicted in figure 11. As hypothesized, when there is more data for the median algorithm to draw upon, the algorithm improves in reducing the overlapping artifacting in the image.

Fig. 11. (Left) the donald_level2 image processed with 6 frames versus (right) the donald_level3 image processed with 12 frames.

To better isolate the problem of alignment, a test case with severe camera shake and some overlap was also executed with a large number of frames. This can be observed in figure 12. While the success metric increased from 64.11% with 6 frames to 64.87% with 12 frames, there is still room for improvement, where alignment was predicted to fix this issue.

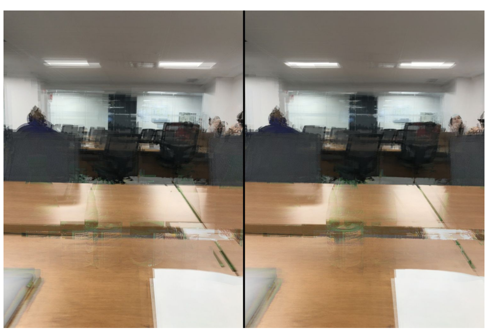

Fig. 12. (Left) shaky_desk processed through regular median filtering with 6 frames versus (right) shaky_desk processed with 12 frames.

D. *Testing the impact of alignment of images*

While increasing the number of frames in the algorithm successfully improves performance, it is not practical to expect for commercial use as a user would be unlikely to take a large number of frames. In Part 1 Section G, possible automation was discussed to remedy this, however implementation was not feasible for this project. Improving alignment of the frames in a sequence, however, was implemented and improves the performance of the median filtering algorithm. It was hypothesized that adding alignment adjustment to the algorithm would eliminate the ghost artifacting that is hallmark of camera shake, and the performance of this addition to the algorithm is discussed in this section.

Figure 13 illustrates the effectiveness of the median filtering algorithm enhanced with alignment for an image with no overlap and severe camera shake. It is clear that the algorithm performs very well, with a metric of 92.53% compared to the regular algorithm's metric of 82.50% for the same test case. Aligning the image eliminates the ghosting artifacts of misaligned sequences and thus makes a more robust image that is more readily usable in a public application.

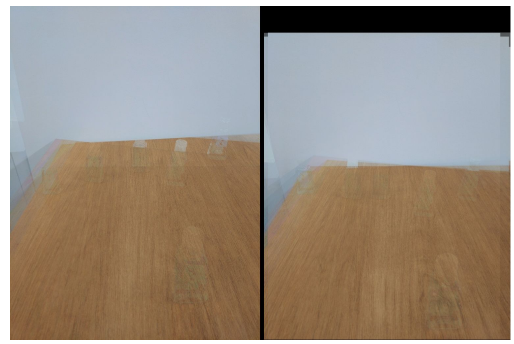

Fig. 13. (Left) sanitizer processed through regular median filtering versus (right) sanitizer processed through pre-aligned median filtering.

The effectiveness of this addition to the algorithm is compounded when analyzed with respect to a test case that includes severe overlap and severe camera shake. Figure 14 demonstrates this, and while the regular median filtering algorithm performed with an effectiveness of 47.02%, the enhanced alignment algorithm performs with an effectiveness of 87.21%.

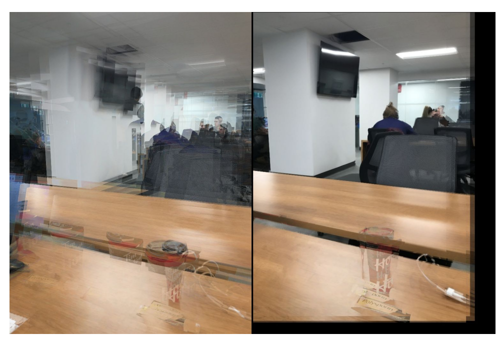

Fig. 14. (Left) tim6 processed through regular median filtering versus (right) tim6 processed through pre-aligned median filtering.

The combination of increased frames and the alignment algorithm can be observed in figure 15 where the sequence features severe camera shake, severe overlap, and a sequence of 12 images. Where the increased data points from the number of frames in the sequence eliminates some of the overlap artifacting (with a success metric of 44.60% processed through regular median filtering), the resulting image is further enhanced with the alignment algorithm, brining the success metric up to 78.91%.

Fig. 15. (Left) tim12 processed through regular median filtering versus (right) tim12 processed through pre-aligned median filtering.

Moreover, the "typical" test case introduced in section # with some camera shake and some overlap performed better with alignment applied than without, increasing from a performance of 66.95% to 81.07%. The resulting image can be observed in figure 16.

Fig. 16. (Left) wallet processed through regular median filtering versus (right) wallet processed through pre-aligned median filtering.

Clearly, the alignment of images with camera shake improves the performance of the median filtering algorithm. The inclusion of this aspect makes the algorithm as a whole more applicable to consumer use and eliminates the need for "perfect" test cases. It should also be noted that alignment of images with no camera shake does not impact the performance of the image at all, considering the images are already aligned.

IV. CONCLUSION

Weighted median filtering proves to be a semi-effective algorithm for eliminating undesirable moving objects across a sequence of images. Without ideal conditions, the algorithm performs sub-optimally, and eliminates moving objects, but leaves a considerable number of artifacts in the resulting image. By employing an alignment algorithm, artifacts left over in the resulting image (leaving a "ghost" artifact) can be improved or eliminated. This leaves the artifacts left behind from overlapping images, something that can be improved by removing the overlapping images or increasing the number of frames used in processing. In employing this technique of improving a sequence of images, the algorithm is effective in producing a satisfactory result that successfully eliminates the moving object from the sequence and produces a clear image of the stagnant features of the scene. As dictated by the proposed metric, the algorithm performed exceptionally well overall when put through both modifications to a regular weighted median filtering algorithm. The quality of the algorithm and program used as a whole is improved by the cropping of alignment bars and colour correction of the resulting image. In this way, the proposed algorithm is better suited for consumer use and is a good candidate for a public application for image processing. As such, the weighted median filtering approach to eliminating objects from sequences of images can be defined as a satisfactory algorithm in achieving its goals.

REFERENCES

[1] David, P. (2013, 29 May). A Look at Reducing Noise in Photographs Using Median Blending. [Article]. Retrieved from https://petapixel.com/2013/05/29/a-look-at-reducing-noise-in-photographs-using-median-blending/

[2] Deng, X., Xiong, Y., Peng, H., "A New Kind of Weighted Median Filtering Algorithm Used for Image Processing," in: Int. Symp. Information Science and Engineering, (2008).

[3] Taguchi, A., "A design method of fuzzy weighted median filters," in: Int. Conf. Image Processing (1996).

[4] Zivkovic, Z. "Improved Adaptive Gaussian Mixture Model for Background Subtraction," in: International Conference on Pattern Recognition, (2004).

[5] Bayar, B., & Stamm, M. C. (2016, June). A deep learning approach to universal image manipulation detection using a new convolutional layer. In *Proceedings of the 4th ACM Workshop on Information Hiding and Multimedia Security* (pp. 5-10). ACM.