CSC320 Assignment 4 Report

Original Noisy Images

My Image (from internet)	Jaguar
	

Control Results of NLM (k=3, w=0.5 * width of image, patch size = 7, h = 25, iterations = 3)

My Image (from internet)	Jaguar
	

Observations:

Fairly significant noise reduction in the blue-noise in the upper part of my image, and a moderate reduction in the amount of RGB noise in the jaguar image.

Explanation

These images are used to compare the subsequent results of this report against so we can observe the impact of certain parameters on the performance of the NLM de-noising algorithm.

NLM Results from Doubling k (3 to 6)

My Image (from internet)	Jaguar
	

Observations

This seems to be the best noise reduction out of any of the variable changes. There blue-section of my image does not appear to be too different from the control, but the lower half is even more de-noised than in the control dataset. However, there does seem to be a small amount of blurring happening near the bloom of the smaller street lights in the photo.

There has been significant improvement in the jaguar’s de-noising. In the control and original image sets the log in the lower half of the photo has a lot of RGB noise. The algorithm has cut this noise down significantly. In addition to this, there was some blurring near the jaguar’s ear/branch by its neck in the control image, however this image seems to have cut down on the amount of blurring

Explanation

The NLM de-noising algorithm relies on a weighted sum of NN pixel values to compute the final color of the de-noised image. Since there is random variation between the noisy pixels, we can reduce the variance of the weighted sum by increasing the number of samples. Intuitively, this makes sense because the different variations “cancel each other out”. Thus, increasing the number of samples (for our case increasing k) available to NLM results in better de-noising results.

NLM Results from Doubling W (half width to full width)

My Image (from internet)	Jaguar
	

Observations

There doesn't seem to be any significant (at least to a human observer) improvements or changes between this image set and the control image set.

Explanation

Perhaps it would be a better idea to change the alpha/decay value, as this would allow us to significantly increase the number of pixels to consider during random search. Doubling w while keeping alpha at ½ only results in random-search considering 1 more pixel. However, if we were to halve alpha for any value, we would effectively increase the number of considered pixels by a factor of 2. Thus, random-search would have a higher probability of increasing the quality of NN matches (as it would effectively be given double the patches to compare). If we had more accurate k-NN matches, the quality of the de-noised image would most likely increase. Since we didn't impact the accuracy of the k-NN field that much by doubling w, we can't expect the quality of the de-noised image to increase too much.

Results from Increasing Patch Size (7 to 9, had to increase h to 31 to compensate)

My Image (from internet)	Jaguar
	

Observations

There doesn't seem to be any significant (at least to a human observer) improvements or changes between this image set and the control image set. In fact, it seems like the blurring described around the Jaguar's neck has increased a little.

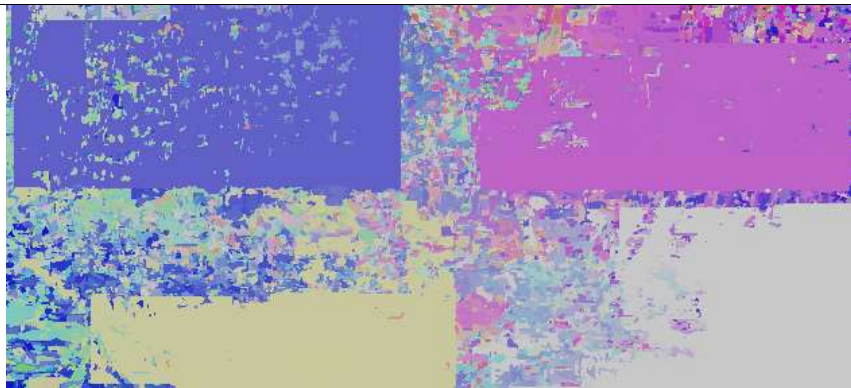
Explanation

Intuitively, increasing the size of the patch should increase the accuracy of the k-NN field. Given more neighbouring pixels, we should be able to make more accurate judgements about whether two pixels are "close" together (in terms of their surrounding pixels). However, in reality it seems like this effect is only marginal - the patch area increased by a factor of 65% (49 to 81) but did not seem produce any noticeable improvements in de-noising. We probably already have a good idea of the "neighbour-ness" of two pixels after we consider a certain number of surrounding pixels, and simply sampling more won't help.

Final Conclusion

It seems that NLM de-noising produces the best results when given a k-NNF where k is as large as possible. Indeed, the algorithm was designed under the assumption that EVERY patch in the image would be considered in order to calculate each de-noised pixel. However, due to the computational difficulty of such a task ($O(P^2)$, where P is the number of patches), this can't be done in any reasonable amount of time. However, only considering the k-best matching patches for every pixel cuts the complexity down to $O(Pk)$, making the algorithm run in an acceptable amount of time. Due to the inverse-exponential weighting function that NLM uses to determine the weighted sum, the worst-matching patches would have a miniscule impact on the final weighted sum when compared to the best-matching patches. Thus, using an (approximate) k-NNF is a reasonable compromise to make in order to make NLM run in an acceptable amount of time.

Final Results of Running K-PatchMatch on Jaguar 2 (k=3, other values default)

Order	NNF Colorized	Reconstruction (judge quality of NNF)
0		
1		
2