Name: Yuhan Shao
Student Number: 1002281327
UTORid: shaoyuha

CSC320 Assignment 4

- **Noisy Jaguar with NLM (source_noise2.png)** **p2-4**
  - **Fix [k=15, iters=3, patch-esize=7], vary w=50, 250, 500 image sets** **p2**
  - **Fix [w=250, iters=3, patch-esize=7], vary k=1, 4, 8, 16 image sets** **p3**
  - **Fix [w=250, k=15, iters=3], vary patch-esize=3, 7, 19 image sets** **p4**

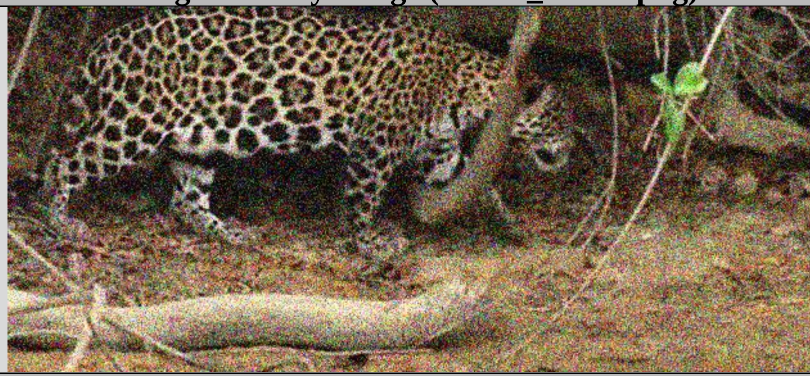
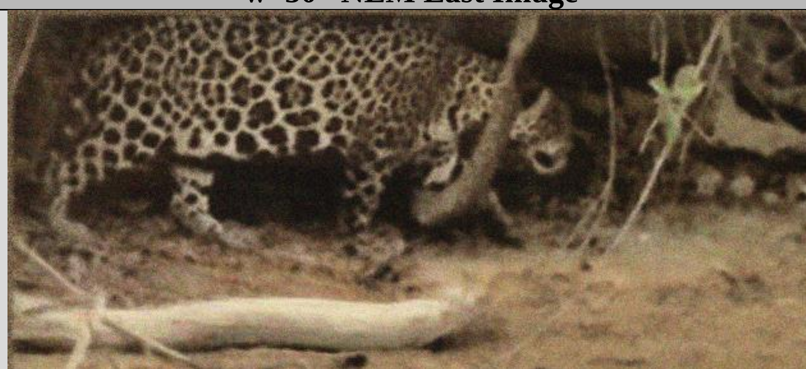
- **Noisy Image with Poor Light Condition with NLM** **p5**
  Online Noisy Image Source: https://en.wikipedia.org/wiki/Image_noise
  - **Fix [k=15, iters=3, patch-esize=7], vary w=50, 250, 500 image sets** **p5**
  - **Fix [w=250, iters=3, patch-esize=7], vary k=1, 4, 8, 16 image sets** **p5**
  - **Fix [w=250, k=15, iters=3], vary patch-esize=3, 7, 19 image sets** **p5**

- **Observation and Explanation** **p6-7**
  - **Fix [k=15, iters=3, patch-esize=7], vary w=50, 250, 500 observation and explanation** **p6**
  - **Fix [w=250, iters=3, patch-esize=7], vary k=1, 4, 8, 16 observation and explanation** **p6**
  - **Fix [w=250, k=15, iters=3], vary patch-esize=3, 7, 19 observation and explanation** **p7**
  - **Conclusion** **p7**

- **Jaguar2 Image pair for k=3 in the Generalized PatchMatch** **p8**

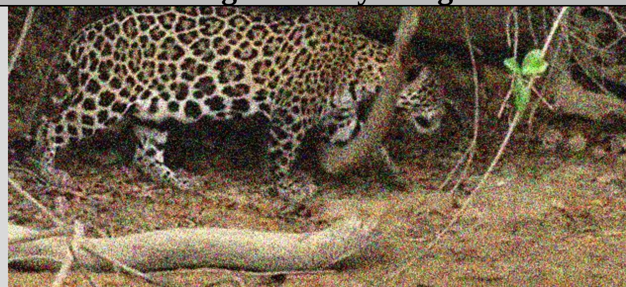
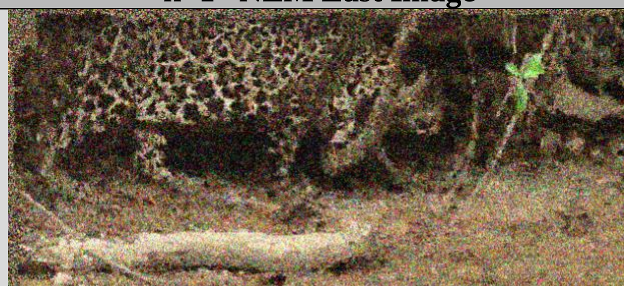
■ Noisy Jaguar with NLM

→ Fix [k=15, iters=3, patch-esize=7], vary w=50, 250, 500

Original Noisy Image (source_noise2.png)

w=50 NLM Last Image

w=250 NLM Last Image

w=500 NLM Last Image


→ Fix [w=250, iters=3, patch-esize=7], vary k=1, 4, 8, 16

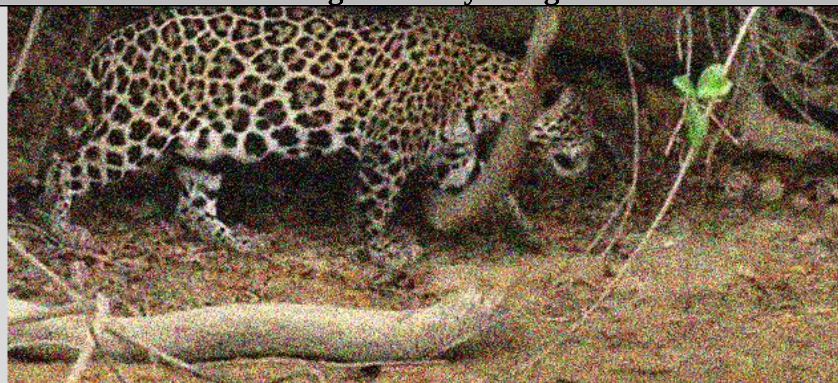
Original Noisy Image

k=1   NLM Last Image

k=4   NLM Last Image

k=8   NLM Last Image

k=16   NLM Last Image


→ Fix [w=250, k=15, iters=3], vary patch-esize=3, 7, 19

Original Noisy Image



patch-esize=3 NLM Last Image

patch-esize=7 NLM Last Image

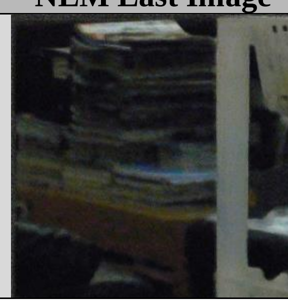
patch-esize=19 NLM Last Image

■ Noisy Image with Poor Light Condition with NLM

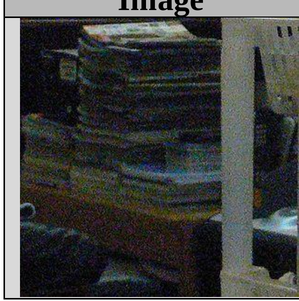
→ Fix [k=15, iters=3, patch-esize=7], vary w=50, 250, 500

Original Noisy Image	w=50 NLM Last Image	w=250 NLM Last Image	w=500 NLM Last Image
			

→ Fix [w=250, iters=3, patch-esize=7], vary k=1, 4, 8, 16

Original Noisy Image			
			
k=1 NLM Last Image	k=4 NLM Last Image	k=8 NLM Last Image	k=16 NLM Last Image
			

→ Fix [w=250, k=15, iters=3], vary patch-esize=3, 7, 19

Original Noisy Image	patch-esize=3 NLM Last Image	patch-esize=7 NLM Last Image	patch-esize=19 NLM Last Image
			

■ Observation and Explanation

- Fix [k=15, iters=3, patch-esize=7], vary w=50, 250, 500

Observation:

When fix k=15, iters=3, patch-esize=7, and vary w (the maximum radius for random search in pixels) with values: 50, 250, and 500, there is no human visible changes or improvement among these three images with increasing w values.

Explanation:

When consider the formula we used in the random search, it is expected to have a better output when increasing the value of w, however, the actual outputs do not have much differences. There may have following two possible reasons: (1) The default w value (the maximum radius for the random search in pixel) is already 250, then when we change to 500, is already the same and has no differences. When w=50, since in our source image there are lots of common part, it is enough for the algorithm to find the best match use w=50. (2) Increasing w but keeping alpha ½ only makes random-search consider one more pixel. However, by contrast, if we consider to half the alpha for ang value, then the number of pixels we considered would be a factor of 2. In this case, there would have higher probability for the random search to increase the quality of NN matches. And when we have more accurate KNN matches, the denoised image's quality would then also increase.
Therefore, since we did not increase the accuracy of the KNN field when increasing the w value, the quality of the denoise image would also not change too much.

- Fix [w=250, iters=3, patch-esize=7], vary k=1, 4, 8, 16

Observation:

This set of variable changes seem to have the best noise reduction compared to the other two sets of variable changes in both the Jaguar noisy images and the poor light condition noisy image. However, we can also observe that even though the image is more denoised when we increase k, the image also becomes blurrier.

Explanation:

By NLM denoising algorithm, we know that the algorithm highly relies on the weighted sum of NN pixel values in order to compute the final pixel color in the denoise image. It is known that there is random variation between the noisy pixels, therefore, we can consider reducing the weighted sum through increasing the number of sample (i.e.: increase the k value). Hence, when we increase the number of samples (i.e.: the k value) available to NLM, we can get a better denoising result.

- Fix [w=250, k=15, iters=3], vary patch-esize=3, 7, 19

Observation:

When fix w=250, k=15, iters=3, and vary patch-esize with values: 3, 7 and 19, there is no human visible changes or improvement among these three images with increasing path-esize values. And the outputs are on the same blur level.

Explanation:

When increase the patch size, the accuracy of the KNN field is supposed to increase. Since when we know more neighbouring pixels, we are supposed to make more accurate result about how two pixels are "close" to each other. However, in our actual denoising output, as the patch size increasing, it seems not produce any human noticeable improvement in the denoising output. This probably because we already know the "distance" between two pixels after we consider a certain number of surrounding pixels, and therefore, when we sample more (i.e.: increase the patch size), it won't help increase the quality of the denoising output image.

- Conclusion

From the above three sets of variables, we can have an idea that NLM denoising algorithm produces the best result when given a KNN field where the value of k is large. In the KNN field algorithm, we only consider the k-best matching patches for each pixel, this largely reduce the running time of the algorithm. In the NLM, it uses the inverse-exponential weighting function in order to determine the weighted sum, therefore, the worst matching patches would get the least weight and decreasing the influences of these worst matching patches in the final output pixel value, by contrast, the best matching patches would be weighted more. Thus, combine KNN field algorithm with NLM is a good choice to denoise a noisy image within a reasonable amount of running time.

■ Jaguar2 Image pair for k=3 in the Generalized PatchMatch

source.png	target.png
	

order0	order1	order2
last_nnf_color	last_nnf_color	last_nnf_color
		
last_reconstruct_source	last_reconstruct_source	last_reconstruct_source
		

last_nnf_vector order 0

last_nnf_vector order 1

last_nnf_vector order 2