1. A description of the non-local means denoising algorithm.

The non-local means (NLM) denoising algorithm was published in 2005 by Antoni Buades, Bartomeu Coll and Jean-Michel Morel. The denoising algorithm removes noise in an image by taking an average of similar non-local neighbourhoods for each pixel in the image. [1]

If p is the observed pixel value, p_0 is the true pixel value (without noise) and n is the value of the noise then the equation $p = p_0 + n$ represents each pixel in the image. If you compute the average of multiple pixels with the same p_0 value the noise goes down proportional to n^2. [3]

Non-local means differs from local means by instead of averaging pixels surrounding a specific pixel (local neighbourhood); the whole image is scanned and an average is taken of the neighbourhoods of every pixel and their neighbourhoods are weighted depending on how similar they are to the original pixel.

$$NL[v](i) = \sum_{j \in I} w(i,j)v(j),$$

Equation 1

From equation 1, $NL[v](i)$ is the estimate pixel value which is the weighted average of all the pixels in the image. Whereby we assume v is the discrete noisy image, $v = \{v(i) \mid i \in I\}$. The weights are between $0 \leq 1$, sum up to 1 and computed via $w(i,j)$. Such that $w(i,j)$ is the weight between pixel i and j. [1]

2. Discussion of the various proposed implementations of the algorithm and discussion of their efficiency. You can start your research on implementation issues from [2].

As proposed in [2] NLM denoising has two main distinct approaches, Pixelwise implementation and Patchwise implementation.

$$\hat{u}_i(p) = \frac{1}{C(p)} \sum_{q \in B(p,r)} u_i(q)\, w(p,q), \qquad C(p) = \sum_{q \in B(p,r)} w(p,q),$$

Equation 2

$$\hat{B}_i = \frac{1}{C} \sum_{Q=Q(q,f) \in B(p,r)} u_i(Q)\, w(B,Q), \qquad C = \sum_{Q=Q(q,f) \in B(p,r)} w(B,Q),$$

Equation 3

A colour image $u = (u_1, u_2, u_3)$ and pixel p in the Pixelwise implementation approach are written in equation 2. Whereby $i = 1,2,3$ and neighbourhood centred at p with a size of $(2r + 1) \times (2r + 1)$ pixels is indicated by $B(p,r)$. $w(p,q)$ represents the weight of the colour neighbourhoods centred at p and q. The weights are calculated using the squared Euclid distance $d^2 = d^2(B(p,f), B(q,f))$ with the neighbourhood size of $(2f+1) \times (2f+1)$.

This implementation works by assigning values to the centre of pixels in a neighbourhood via the average of all similar pixels.

On the other hand, the Patchwise implementation is expressed in equation 3. Patchwise has the same formulation as the Pixelwise implementation however the weighting is calculated in patches, $w(B(p,f), N(q,f))$, rather than individual pixels.

As a result, this disposes of $N^2 = (2f+1)^2$ possible estimates for each pixel by calculating all the patches in the image. Averaging these estimates will produce the final denoised image. [1]

As highlighted in [4] the Pixelwise NLM implementation is faster than the Patchwise implementation by a factor of 6 to 49.

Since the NLM denoising scans the whole image it is therefore computationally expensive with a computational complexity of $O(n^4)$. This is because the Pixelwise implementation is $O(n^2 \times (2f+1)^2)$ for every pixel with n representing the image width and $(2f+1)^2$ is the width of the neighbourhood. This is equivalent to $O(n^2 \times n^2 \times (2f+1)^2) = O(n^4)$. Since the Patchwise implementation disposes of $(2f+1)^2$ in each patch calculation the time complexity per pixel is just $O(n^2)$. But since $(2f+1)^2$ is constant the time complexity is $O(n^4)$.

The usage of FFT (Fast Fourier Transform) speeds up computation time in implementing NLM methods [08].

The main difference of both implementations is the gain on PSNR (peak signal to noise ratio) by the patchwise implementation, due to the larger noise reduction of the final aggregation process. But with regards to the preservation of detail in the patchwise implementation, there is no change in overall quality. [2].

3. Demonstration of the influence of the algorithmic parameters on the output.

I will use *OpenCV* to demonstrate the different parameters of the NLM algorithm. There are 3 parameters which influence the output of the algorithm: *templateWindowSize*, *searchWindowSize* and the *filtering parameter (h)*.

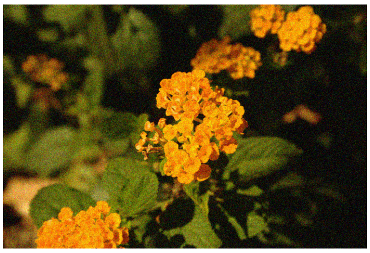
Firstly we will discuss the *templateWindowSize*, which is the size of the window that is used in the calculation of the weights. This is in terms of pixels and is recommended to be odd and 7 (7x7 pixels). 7x7 is recommended for the templateWindowSize due to being shown to be large enough to be tolerant of noise and not sacrifice fine detail or structure. The higher the templateWindowSize the more robust to noise the denoising algorithm will be at the cost of detail as shown in 1.a, 1.b and 1.c. (see appendix for larger images).

Original Image with noise (1.a)

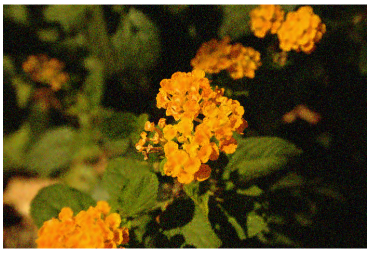
templateWindowSize = 5 (1.b)

templateWindowSize = 11 (1.c)

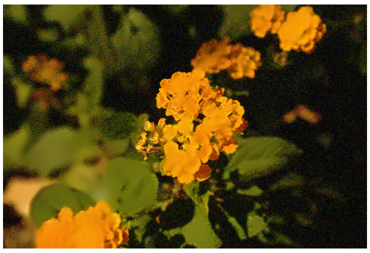
Secondly, is the *searchWindowSize* parameter. This parameter controls the size of the neighbourhood used in the calculation of the weights by imposing limits on the maximum distance in pixels for which similar pixels can be found. This is recommended as well to be an odd number [1]. At the cost of sharp edges and fine detail, increasing the value of searchWindowSize would increase the robustness of the output to noise as shown in 2.a, 2.b & 2.c.



Original Image with noise (2.a)



searchWindowSize = 7 (2.b)



searchWindowSize = 35 (2.c)

Finally, the last parameter h determines the intensity of filtering. The value can be expressed as $h = k\sigma$. Whereby k is a programmable constant and σ is the noise standard deviation. A low value of h would maintain the fine details but would result in reduced noise removal meanwhile on the opposite spectrum if h is too high the majority of noise would be removed but sacrifices the fine details resulting in a blurrier image as shown in 3.a & 3.b.

h = 5 (3.a)

h = 15 (3.b)

4. Discussion of the strengths and limitations of non-local means compared to other denoising algorithms.

Other denoising algorithms including median filtering and Gaussian smoothing are examples of local neighbourhood denoising algorithms. These ‘local’ denoising algorithms generally perform poorly compared to ‘non-local’ denoising algorithms due to pixels nearby in the neighbourhood are not very similar and which therefore does not produce as an effective estimation. The non-local denoising algorithms work particularly well in pictures which have patterns or textured due to non-local neighbourhoods having similar patches within the image itself. [1]

Anisotropic filtering [1] works well to restore straight edges however flat and textured patches are degraded. It is sometimes used instead of the Gaussian smoothing as it avoids the blurring effect by convolving the image u at *x* only in the direction orthogonal to *Du(x)*. [1] [9]

Non-local means implementations disadvantages compared to local denoising algorithms is that due to scanning the whole image, the time complexity is increased by the number of pixels in the image requiring more computation. Unfortunately, the method noise introduced when using NLM implementations is similar in style to white noise. [1]

5. Description and discussion of modifications and extensions of the algorithm that have been proposed in the literature.

There are many modifications and extensions to the NLM algorithm in literature due to the popularity and great results when implemented.

From reference [7] an extension called Non-Local Bayes improves the NLM by using a Gaussian vector model on similar patches within the image. A covariance matrix is introduced which estimates the variability of said patches which when inverted produces an optimal estimate of each noisy patch in the set. Every patch not only has a mean associated with it but also a covariance matrix too.

Another modification of NLM is Fast Non-local Algorithm proposed by Jin Wang and co [8]. This method greatly reduces the computational cost in calculating the similarity of neighbourhood windows. Both summed square image (SSI) scheme and fast Fourier transform (FFT) is used to accelerate the computation time resulting in a ‘fifty times faster than the original non-local algorithm’ but also produces on par perceptual image quality and comparable mean-squared error compared to the original approach.

6. Description and discussion of applications of the original algorithm and it's extensions.

There are many applications of the NLM denoising algorithm. An example is [9] Optimising NLM for denoising low dose CT. This is used to increase the peak signal to noise (PNSR) when denoising low dose phantom CT images.

Another use of NLM is in Diffusion tensor imaging (DT-MRI). Denoising the resulting images helps to increase the quality of estimated tensor fields. This is important as DT-MRI is particularly sensitive to image noise. The denoised images provide the user with better image interpretation and enhanced image quality. Non-local Means produced a better result than other methods such as Gaussian Smoothing, Anisotropic Diffusion and Total Variation. [10]

Finally, another application of Non-local means is to denoise ultrasound images. The resulting images have smoothed homogeneous areas whilst keeping clear contours which aid in extracting information from the images. NLM algorithms are also very efficient in denoising images and fast which is particularly useful in denoising ultrasound live videos in video analysis, visual servoing or image-guided surgical interventions which all require realtime image denoising. [11]

References:

[1] Buades Antoni, Bartomeu Coll, and J-M. Morel. "A non-local algorithm for image denoising." 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition *(CVPR'05)*. Vol. 2. IEEE, 2005.

[2] Buades, Antoni, Bartomeu Coll, and Jean-Michel Morel. "Non-local means denoising." *Image* Processing On Line 1 (2011): 208-212.

[3] Alan Saberi. https://www.youtube.com/watch?v=9tUns4HYtcw&fbclid=IwAR19nOUG2Yqw-RD_jRKyUcmzEBYpr-yRuP2YNBRtwtNKLR4_gO13QAljvvU

[4] Froment, Jacques. (2014). Parameter-Free Fast Pixelwise Non-Local Means Denoising. Image Processing On Line.

[5] Zhang, H., Zeng, D., Zhang, H., Wang, J., Liang, Z. and Ma, J. (2017), Applications of nonlocal means algorithm in lowdose Xray CT image processing and reconstruction: A review. Med. Phys., 44: 1168-1185.

[6] P. Perona and J. Malik, Scale space and edge detection using anisotropic diffusion, IEEE Trans. Pant. Anal. Mach. Intell., 12, pp. 629-639, 1990.

[7] Marc Lebrun, Antoni Buades, and Jean-Michel Morel, Implementation of the “Non-Local Bayes" (NL-Bayes) Image Denoising Algorithm , Image Processing On Line , 3 (2013), pp. 1–42.

[8] Wang, Jin; Guo, Yanwen; Ying, Yiting; Liu, Yanli; Peng, Qunsheng (2006). ”Fast non-local algorithm for image denoising”. International Conference on Image Processing. pp. 14291432.

[9] Zachary S. Kelm, Daniel Blezek, Brian Bartholmai, Bradley J. Erickson (2009). “Optimizing non-local means for denoising low dose CT”. 2009 IEEE International Symposium on Biomedical Imaging: From Nano to Macro

[10] Nicolas Wiest-Daesslé, Sylvain Prima, Pierrick Coupé, Sean Patrick Morrissey, Christian Barillot. “Non-Local Means Variants for Denoising of Diffusion-Weighted and Diffusion Tensor MRI”. Part of the Lecture Notes in Computer Science book series (LNCS, volume 4792).

[11] Fernanda Palhano, Xavier de Fontes, Guillermo Andrade Barroso, Pierrick Coupé, Pierre Hellier (2010). “Real time ultrasound image denoising”. March 2011, Volume 6, Issue 1, pp 15–22

Appendix:

Original Image with noise (1.a)

templateWindowSize = 5 (1.b)

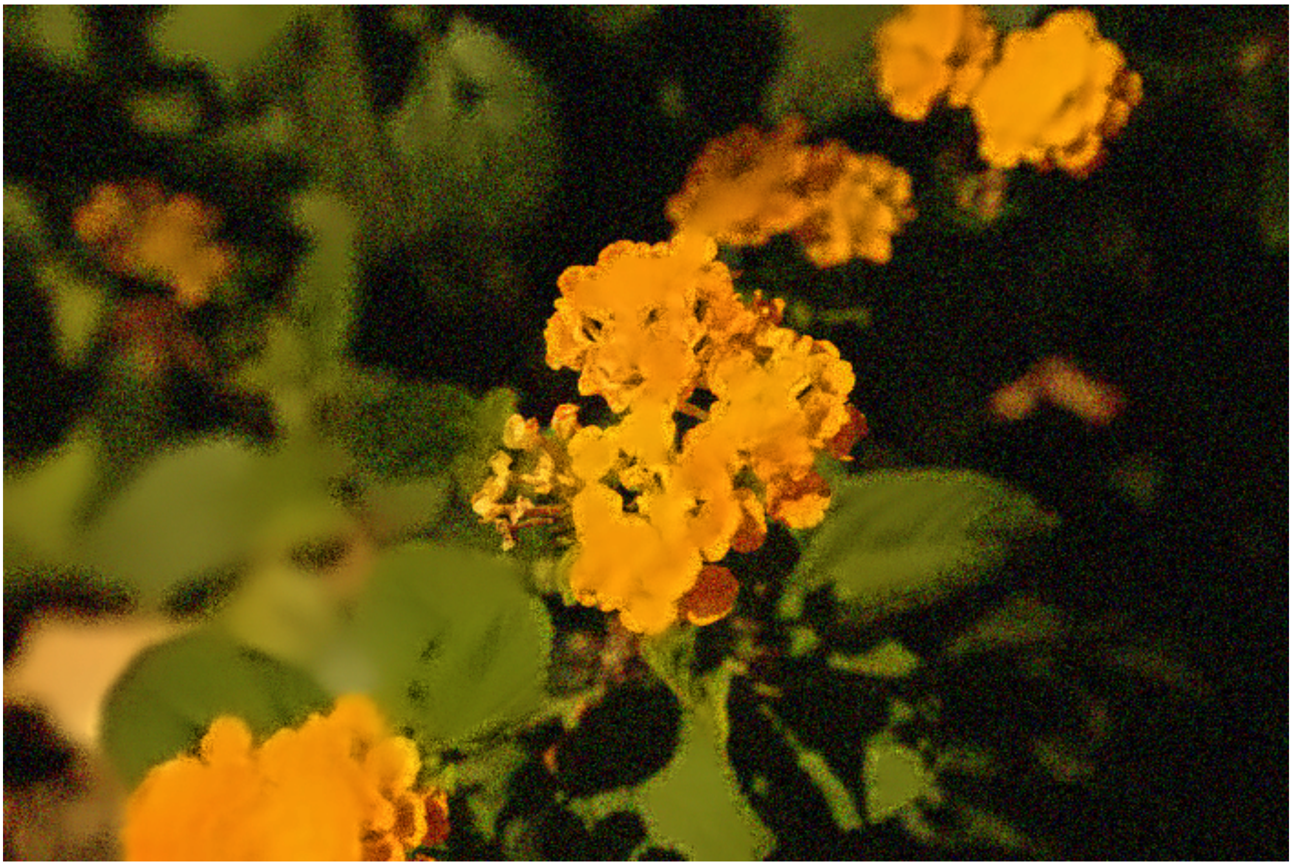
Appendix:

templateWindowSize = 11 (1.c)

Original Image with noise (2.a)

Appendix:

searchWindowSize = 7 (2.b)



searchWindowSize = 35 (2.c)

Appendix:

h = 5 (3.a)

h = 15 (3.b)