Non Local Means

Dinesh Subhuraaj, Eram Arfa, Priyashi Yadav

Institut fur Parallele und Verteilte Systeme, IPVS, University of Stuttgart

1. **INTRODUCTION**

**Non-local means** is an image processing algorithm for [image denoising](https://en.wikipedia.org/wiki/Image_denoising). Local mean filters, take the [mean](https://en.wikipedia.org/wiki/Mean) value of a group of pixels surrounding a target pixel to smoothen the image, whereas non-local means filtering takes mean of all pixels in the image, weighted by how similar these pixels are to the target pixel. This results i much greater post-filtering clarity, and less loss of detail in the image compared with local mean algorithms. If compared with other well-known denoising techniques, non-local means adds "method noise" (i.e. error in the denoising process) which looks more like [white noise](https://en.wikipedia.org/wiki/White_noise), which is desirable because it is typically less disturbing in the denoised product. Recently non-local means has been extended to other image processing applications such as [deinterlacing](https://en.wikipedia.org/wiki/Deinterlacing), view interpolation, and depth maps regularization.

The [computational complexity](https://en.wikipedia.org/wiki/Computational_complexity) of the non-local means algorithm is quadratic in the number of pixels in the image, making it particularly expensive to apply directly.

1. **DEFINITION**

Suppose {\displaystyle \Omega }ῼ is the area of an image, and *{\displaystyle p}p* and *{\displaystyle q}q* are two points within the image. Then, the algorithm is



where *u(p*){\displaystyle u(p)} is the filtered value of the image at point *p* {\displaystyle p},*v*(*q*) {\displaystyle v(q)} is the unfiltered value of the image at point q{\displaystyle q}, f(p,q) {\displaystyle f(p,q)} is the weighting function, and the integral is evaluated  q {\displaystyle \forall q\in \Omega }.

*C*(*p*) {\displaystyle C(p)}is a normalizing factor, given by:



1. IMPLEMENTATION

3.1. Implementation on CPU

3.2. Implementation on GPU

1. RESULTS
2. CONCLUSION
3. TASKS DONE BY

Dinesh

Eram

Priyashi

1. REFERENCES