ADVANCED IMAGE ANALYSIS
BILATERAL FILTERING

Presented By: Chalikonda Prabhu Kumar
(ChaKon)

Contents

Contents

Introduction to Filtering

- AIM: Smoothing image to reduce noise
- Examples of Filtering like Mean, Gaussian and Median Filtering.

Figure: Original

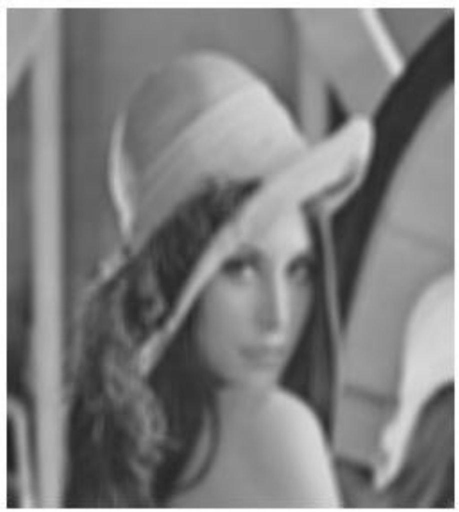


Figure: Mean, radius $6px$

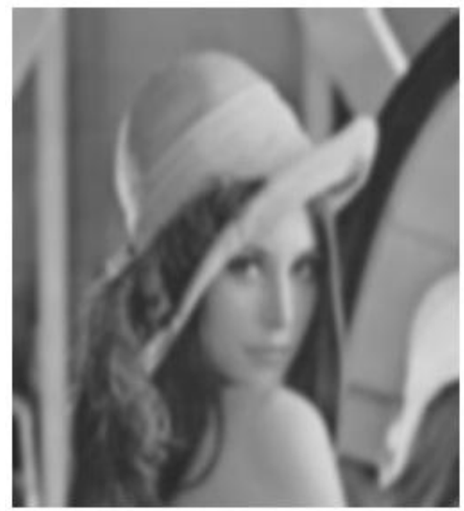


Figure: Gaussian, $\sigma = 4.0$

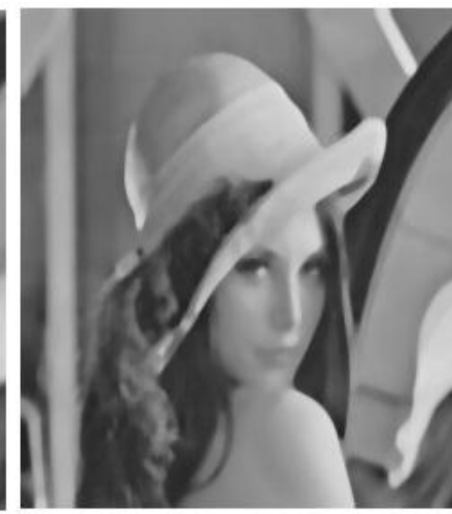


Figure: Median, radius $6px$

Contents

- Introduction to Filtering
- Problems in Filtering
- Bilateral Filtering
- Algorithm & Implementation
- Results
- Applications
- Conclusion

Problems in Filtering

- Mean Filtering : Blurs the image, removes simple noise and no details are preserved
- Gaussian Filtering : Blurs the image, results related to mean filter, preserved details for small sigma values
- Median Filtering : Preserves some details, good in removing strong noise.

Contents

- Introduction to Filtering
- Problems in Filtering
- Bilateral Filtering
- Algorithm & Implementation
- Results
- Applications
- Conclusion

Bilateral Filtering

- Aim : Need to smooth regions but not to smooth edges.

- This is non iterative, local and simple.

- Smoothing should done as usual in the domain of image.

Bilateral Filtering

- Should not smooth when pixels are not similar to preserve edges.

- Bilateral image produce no phantom colors along edges in color images, and reduce phantom colors where they appear in the original image.

Contents

Algorithm & Implementation

Algorithm Idea :

- Smooth as usual in the domain of the image (E.g. Gaussian)
- Do not smooth when pixels are not similar (Preserving Edges)

Algorithm & Implementation

Similarity Function :

- Determines the amount of smoothing
  - Similar Pixels → Strong smoothing
  - Otherwise (edges) → No Smoothing

- Similarity is based on human perception.
  Based on the intensity values of the pixel.

Algorithm & Implementation

- Domain Filter + Range Filter → Bilateral Filtering
- $J = \frac{1}{k_s} \sum_{p \in \Omega} f(p-s)\, I_p$ $\quad k = \sum_{p \in \Omega} f(p-s)$ + $J = \frac{1}{k_s} \sum_{p \in \Omega} g(I_p - I_s)\, I_p$ $\quad k = \sum_{p \in \Omega} g(I_p - I_s)$ = $J_s = \frac{1}{k_s} \sum_{p \in \Omega} f(p-s)\, g(I_p - I_s)\, I_p$ $\quad k_s = \sum_{p \in \Omega} f(p-s)\, g(I_p - I_s)$
- *s* coordinate of center pixel, *p* coordinate of current pixel, Ω set of all pixels in local neighbor pixels.
- J_s is resulting pixel intensity. I_s, I_p intensities of *p* & *s*.
- *f(p-s)* measures the geometric distance between *p* & *s*.
- $g(I_p - I_s)$ measures photometric similarity between I_p & I_s.

Algorithm & Implementation

- Domain Weighting function
  - It is standard Gaussian filter.

$$f\,(p-s) = e^{-\frac{d(p-s)^2}{2\sigma_d^2}}$$

$$d\,(p-s) = \|p-s\| = \sqrt{p^2+s^2}$$

d(*p*-*s*) is Euclidean distance between *p* & *s*.

  - If we increase the larger the domain parameter it blurs the image. It should choose based on the desired amount of low pass filtering.

Algorithm & Implementation

- Range Weighting function
  - It is Gaussian intensity difference

$$g(I_p - I_s) = e^{-\frac{\delta(I_p - I_s)^2}{2\sigma_r^2}}$$

$$\delta(I_p - I_s) = \|I_p - I_s\|$$

$||I_p - I_s||$ measures the difference between two pixels.

  - If the image is amplified or attenuated range parameter should adjust.

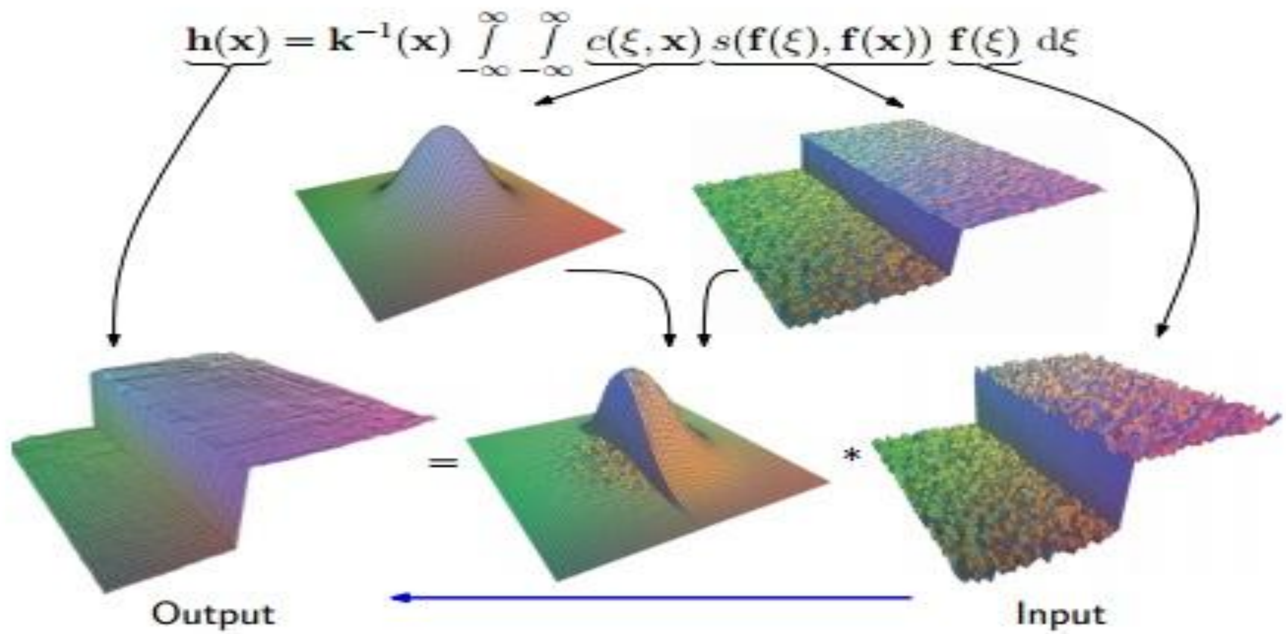
Algorithm & Implementation

(a) (b) (c)

- When bilateral filter is centered on pixel on bright side of the boundary, the similarity function assumes close to 1 on same side and 0 for pixels on dark side.
- The similarity function for window chosen in fig(b). The normalizer *Ks* weights for all pixels adds up to 1. As a result bright pixels at center replaced by average of bright pixels and dark pixels are ignored.
- Good filtering by domain component and edges are preserved by range component fig (c).

Algorithm & Implementation

- Big Picture



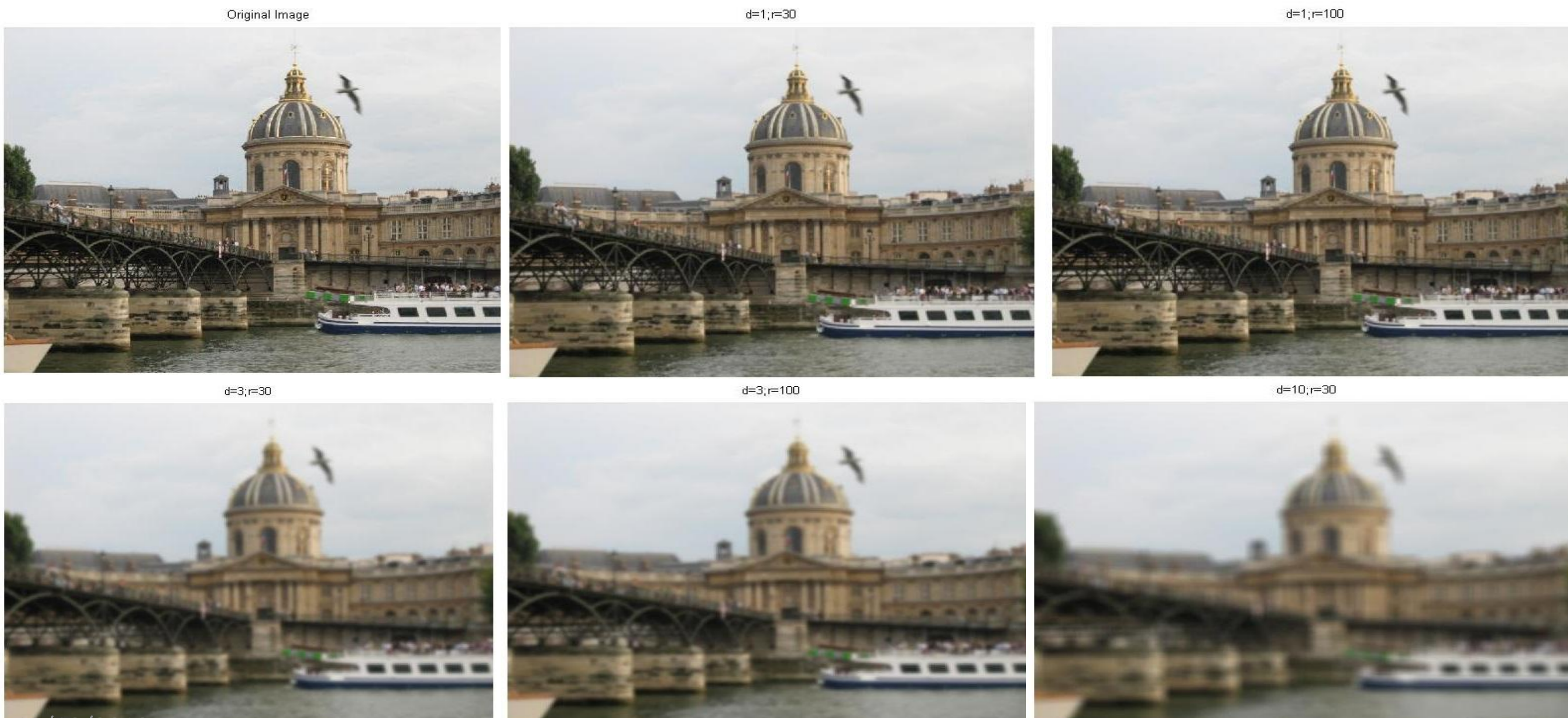
Contents

Results

- Result images for different range and domain values.

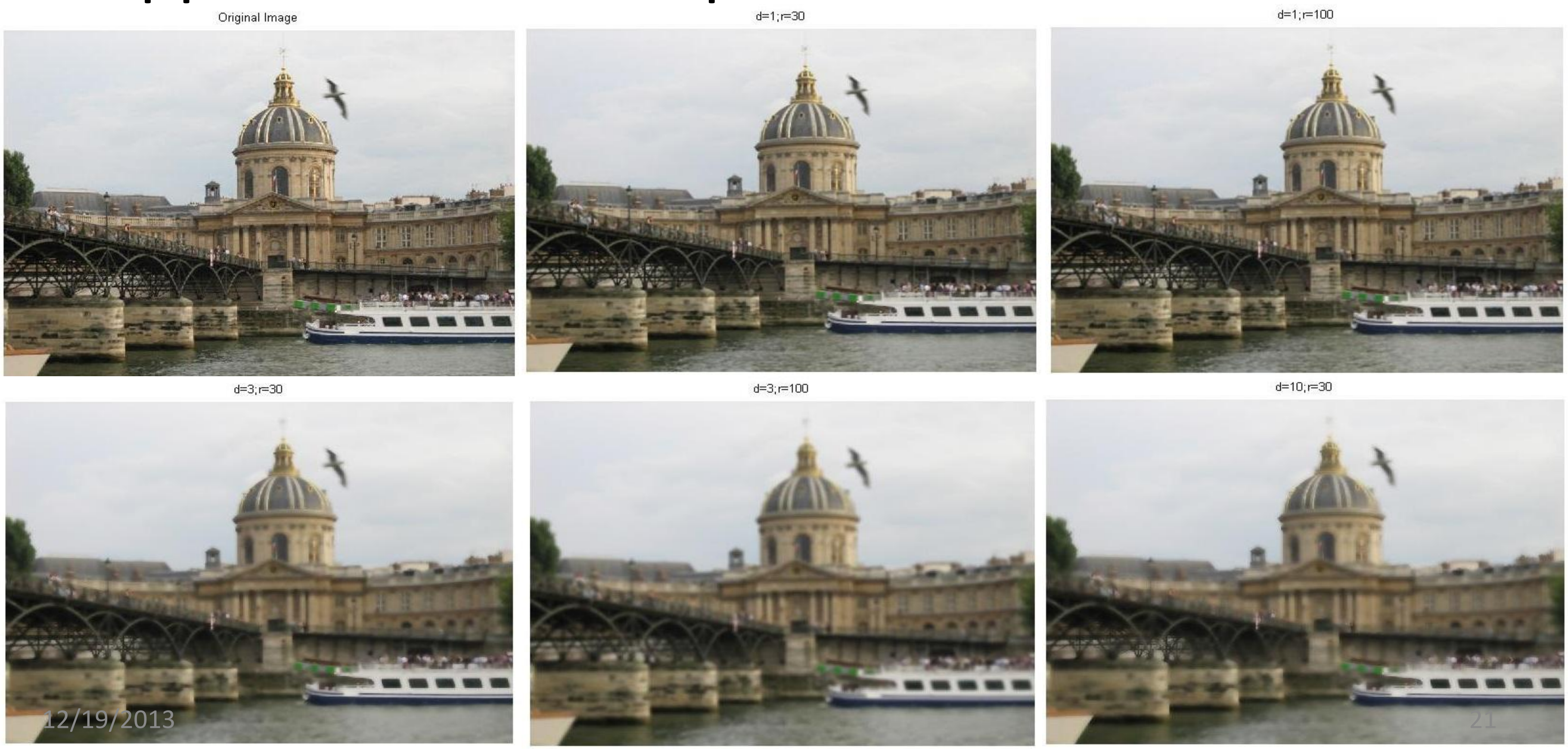
Results

- Results for color images bilateral filter applied on each channel.



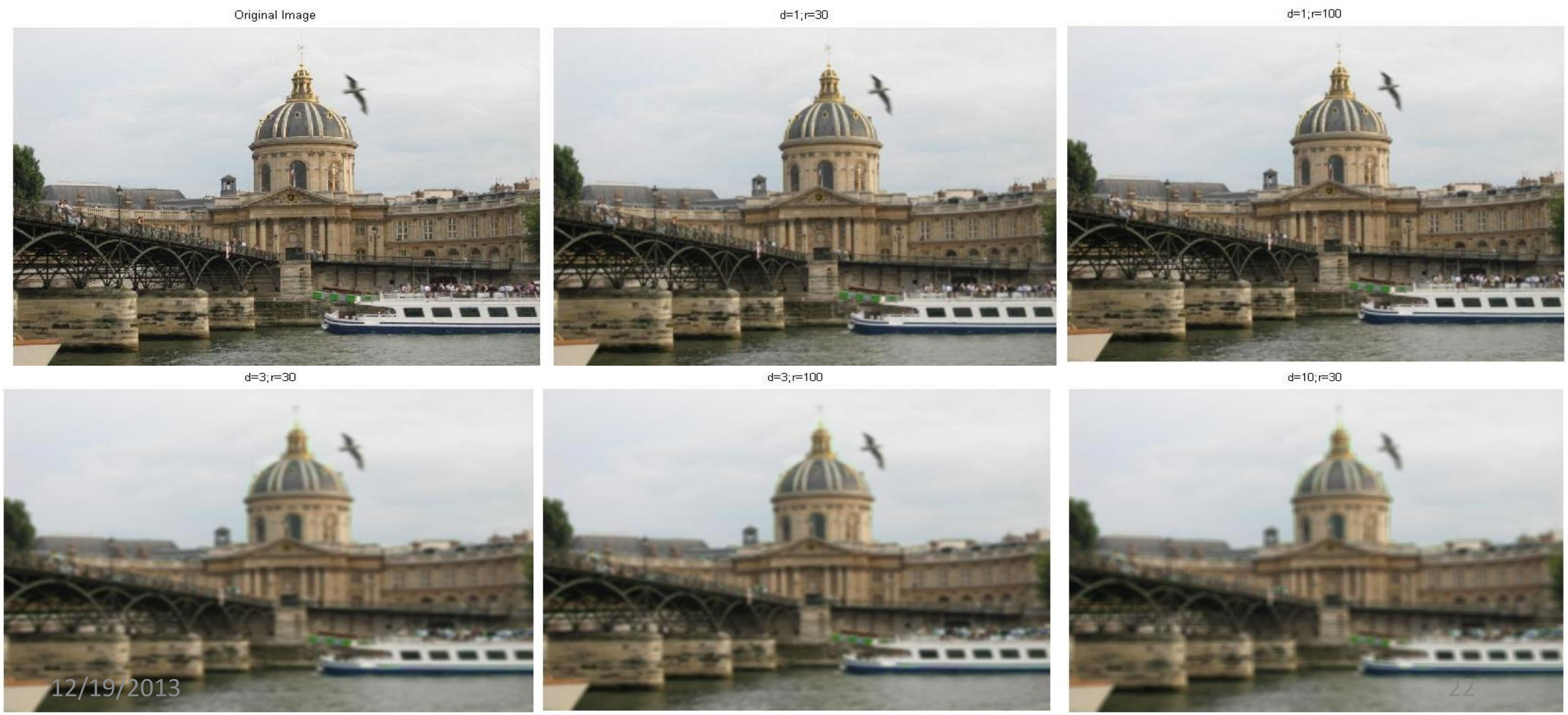
Results

- Results for color images when bilateral filter applied on CIE-Lab space.



Results

- Results for color images when bilateral filter applied on HSV color space.



Experiments

- Adding Gaussian noise to image.

Contents

- Introduction to Filtering
- Problems in Filtering
- Bilateral Filtering
- Algorithm & Implementation
- Results
- Applications
- Conclusion

Application

- De-noising
- Picture simplification
- Contrast reduction
- Mesh smoothing and many more..

Contents

- Introduction to Filtering
- Problems in Filtering
- Bilateral Filtering
- Algorithm & Implementation
- Results
- Applications
- Conclusion

Conclusion

- Bilateral filtering is best for preserving more information (edges).
- CIE Lab color space gives better output than bilateral filter applied on each channel.
- Parameters of domain filter depends on image properties.
- Details are lost with large range values but edges are preserved at all ranges scales that are below the maximum image intensity value.
- Hard to analyze bilateral filtering because of non linear nature.

References

- Bilateral filtering for Gray and Color Images by C.Tomasi & R.Manduchi
- Course Slides
- www.google.com


THANK
YOU !