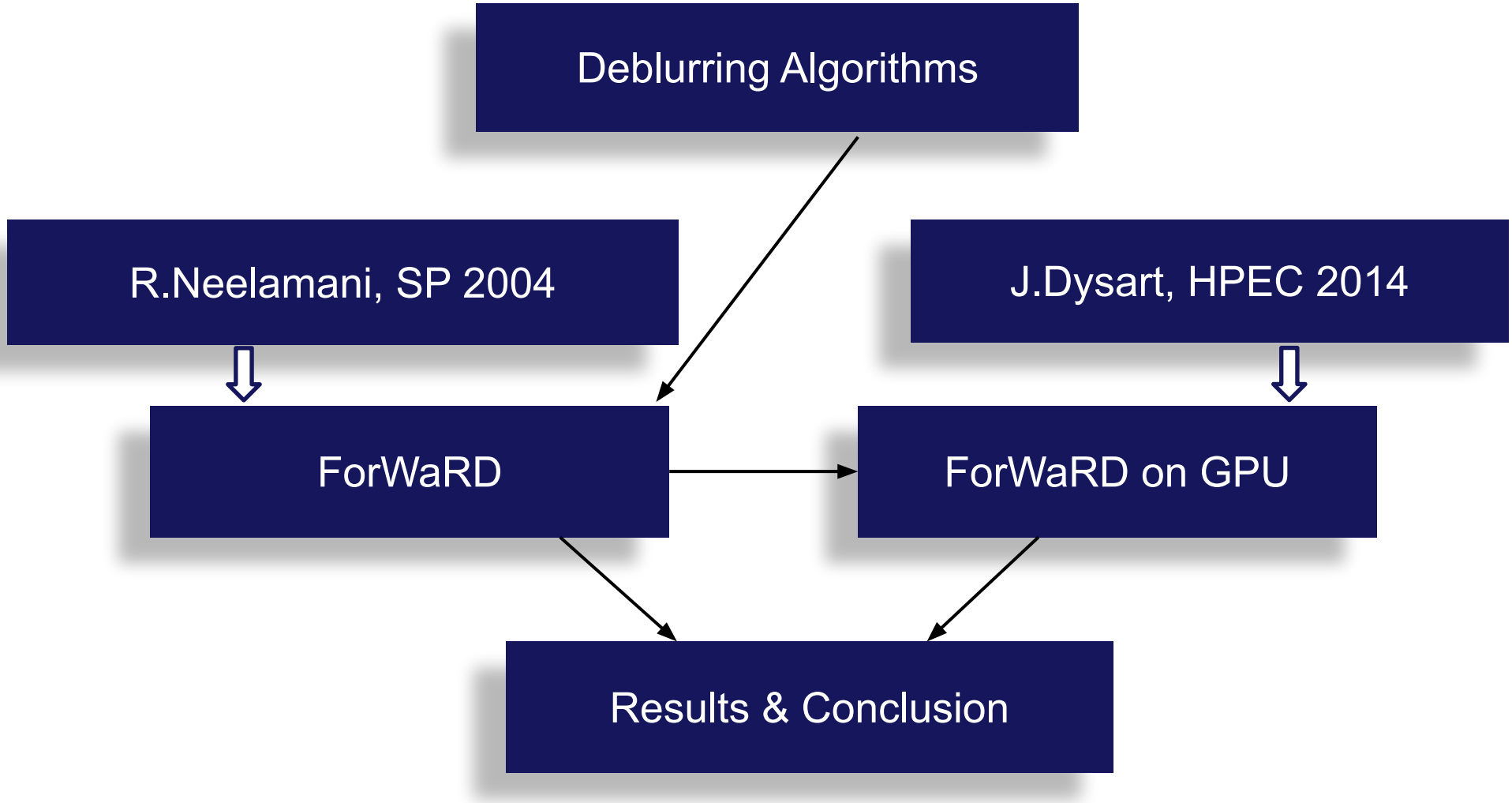
Fourier-Wavelet Regularized Deconvolution (ForWaRD)

Seminar: Embedded Image Processing

Professor: Simon **Supervisor**: Yousef Baroud

Outline


Deblurring Algorithms
R.Neelamani, SP 2004
J.Dysart, HPEC 2014
ForWaRD
ForWaRD on GPU
Results & Conclusion

Introduction

Original Image

⊗ PSF ✚ AWGN =

Blurred Image

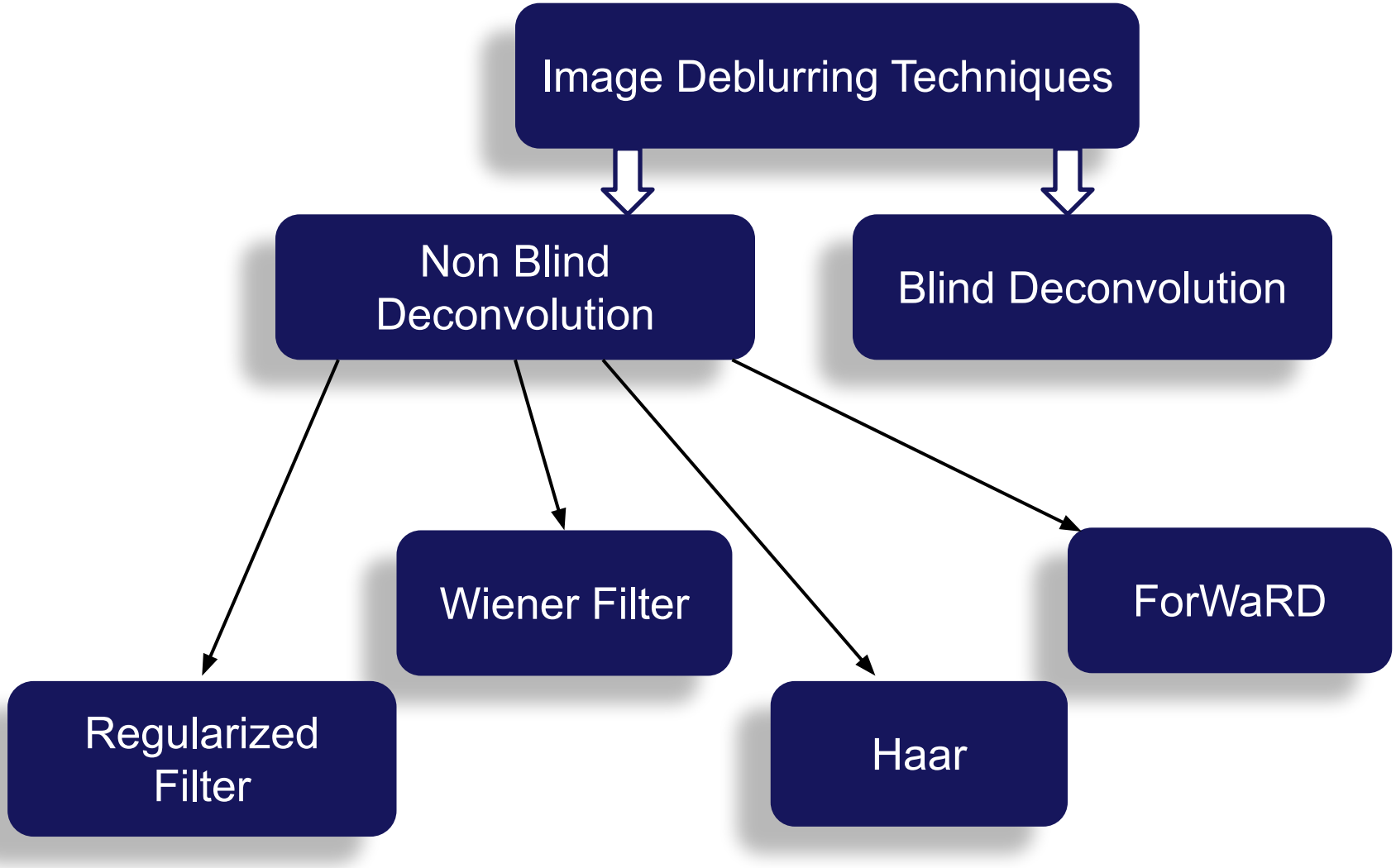
PSF- Point Spread Function
AWGN - Additive White Gaussian Noise

Motivation

- Image blur Model: f (x,y)*g(x,y) + n(x,y) = h(x,y)
- Naive Deconvolution (Inverse fourier transform)
  - 1/G(w)
  - Limitation: Large Mean Square error
- How to estimate the original image?

Deblurring Algorithms

Deblurring Algorithms



Wiener Deconvolution

- Wiener Filter $\dfrac{G^*(\omega)}{|G(\omega)|^2 + \dfrac{S_n(\omega)}{S_f(\omega)}} \Rightarrow \dfrac{G^*(\omega)}{|G(\omega)|^2 + K}$

- J = deconvwnr(I,PSF,NSR)

Ref[3]: Digital Image Processing, C. Gonzalez

Wiener Filter Result

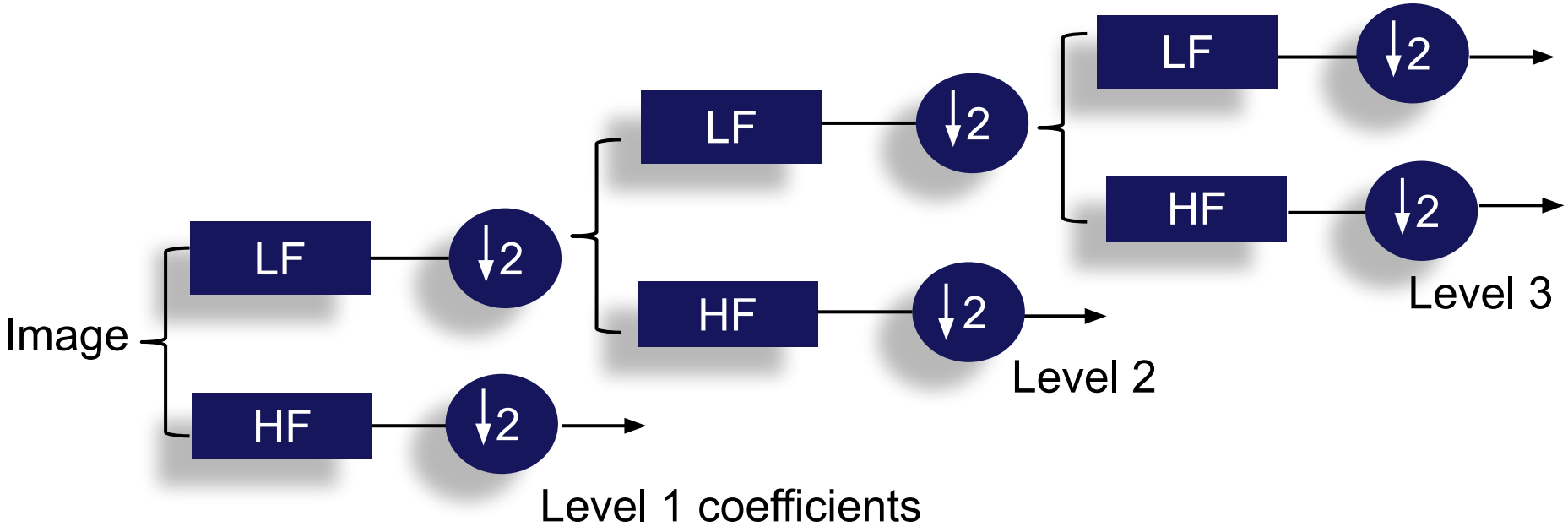
Original image

Noisy image, SNR =40dB

Wiener, ISNR =5.61dB, SNR =20.79dB

- \+ Sharp images
- \- Singularities

Wavelet Domain



HF - High pass Filter

LF - Low pass Filter

2 Down sampling

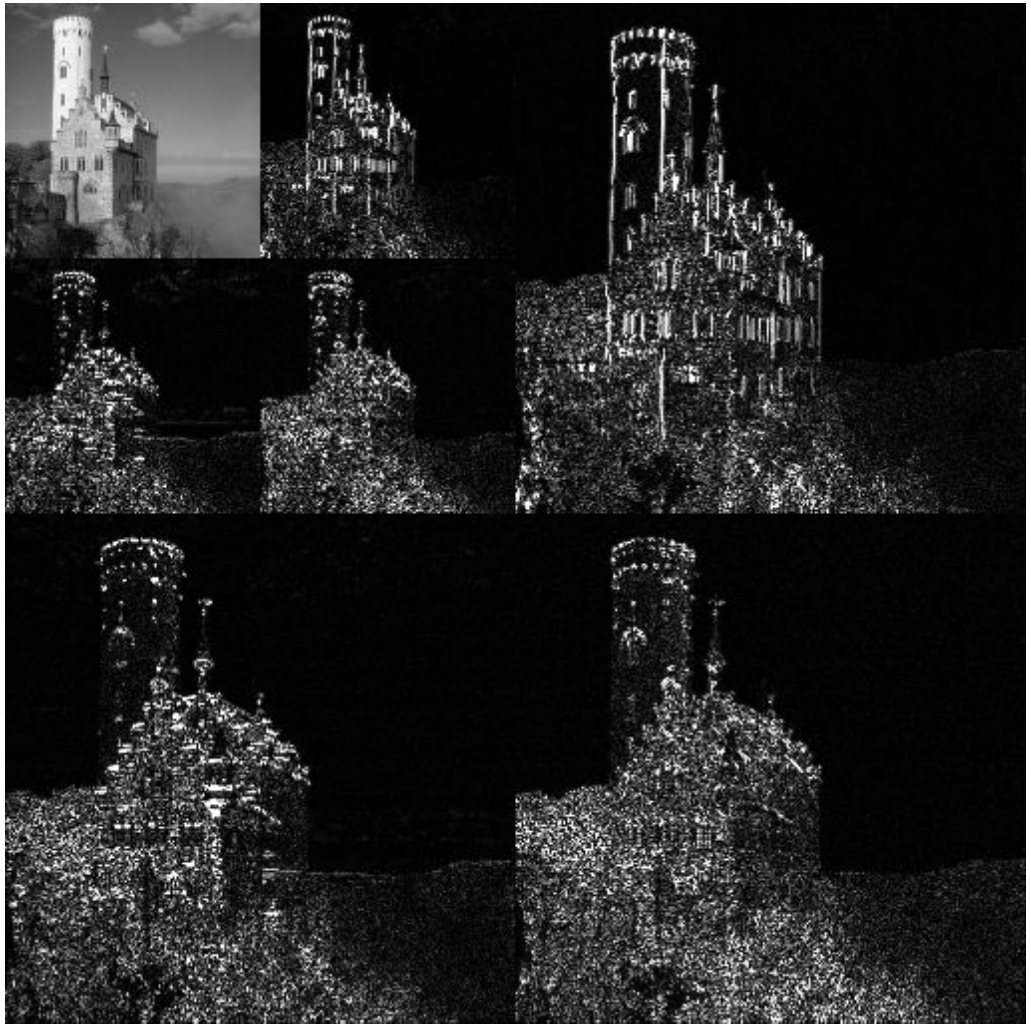
Haar Wavelet Transform

- Forward Transform
  - At each level
    - Scaling coefficients
      - Avg. of adjacent samples
    - Wavelet coefficients
      - Sub. of adjacent samples

6	8	6	8
3	6	3	1
2	4	2	3
4	6	4	3

- [ca,ch,cv,cd]=Dwt2(X, ‘wname’)

Wavelet Transform Result



- \+ Smooth regions
- \- Noise

ForWaRD Algorithm

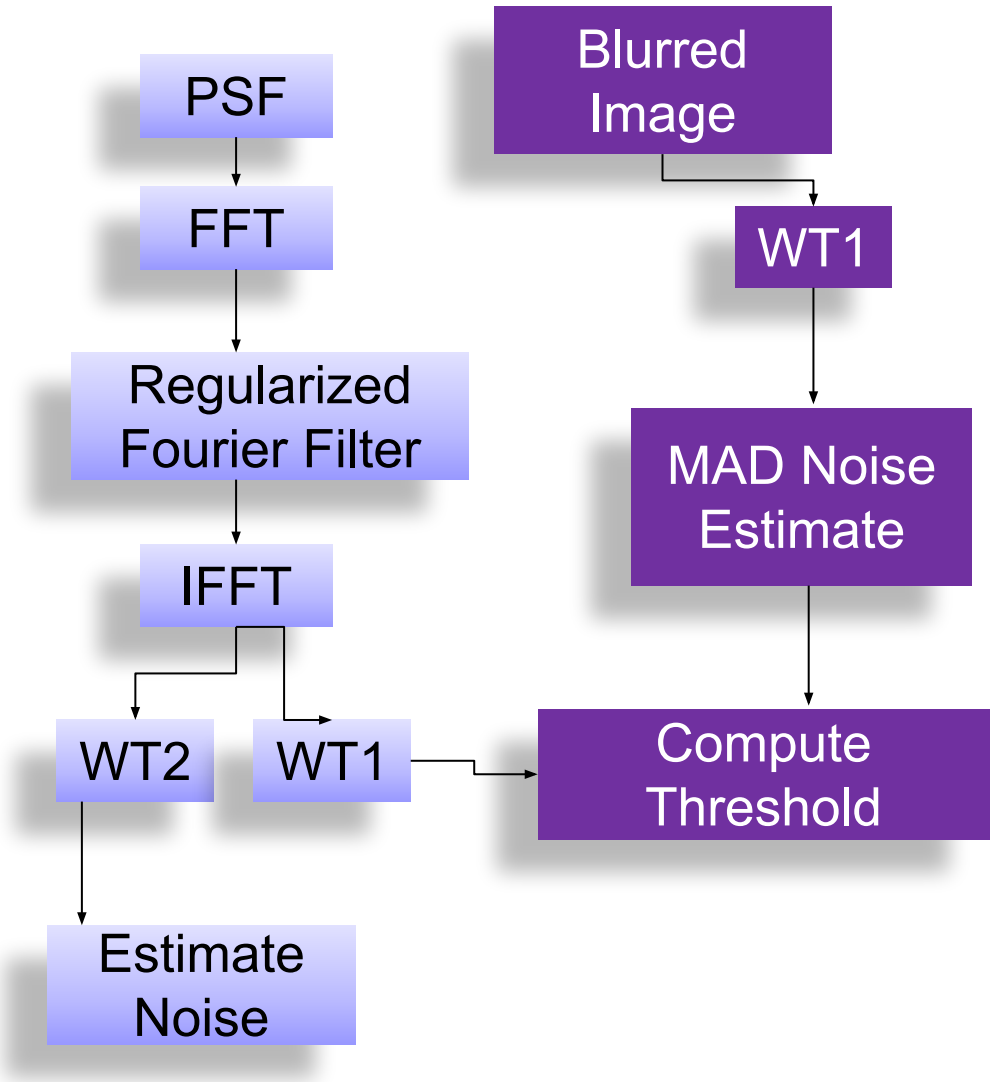
- Fourier Shrinkage
  - Tikhonov filter, K= $3.4*10^{-4}$

$$\frac{G^*(\omega)}{|\,G(\omega)\,|^2 + K}$$

- Wavelet Shrinkage
  - Coefficients Threshold : Y = wthresh(X,SORH,T)
  - Wavelet domain wiener filter

Ref[1]: R. Neelamani, 'Forward for ill conditioned systems'

ForWaRD Implementation



Precompute

Threshold Update

PSF - Point Spread Function

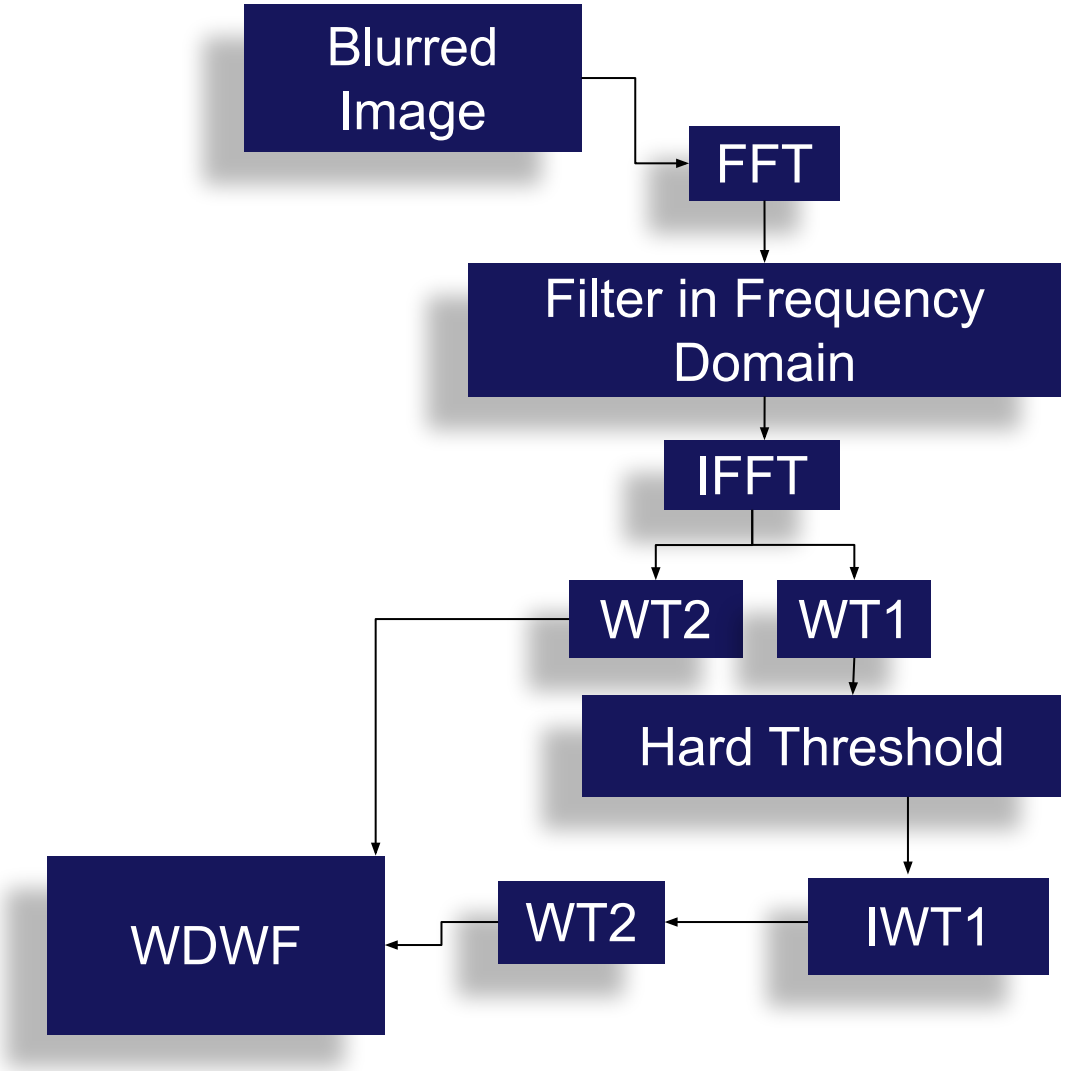
FFT - Forward Fourier Transform

IFFT - Inverse FFT

WT1, WT2 } Daubechies6, 2 Wavelet Transforms

MAD - Median Absolute Deviation

ForWaRD Implementation

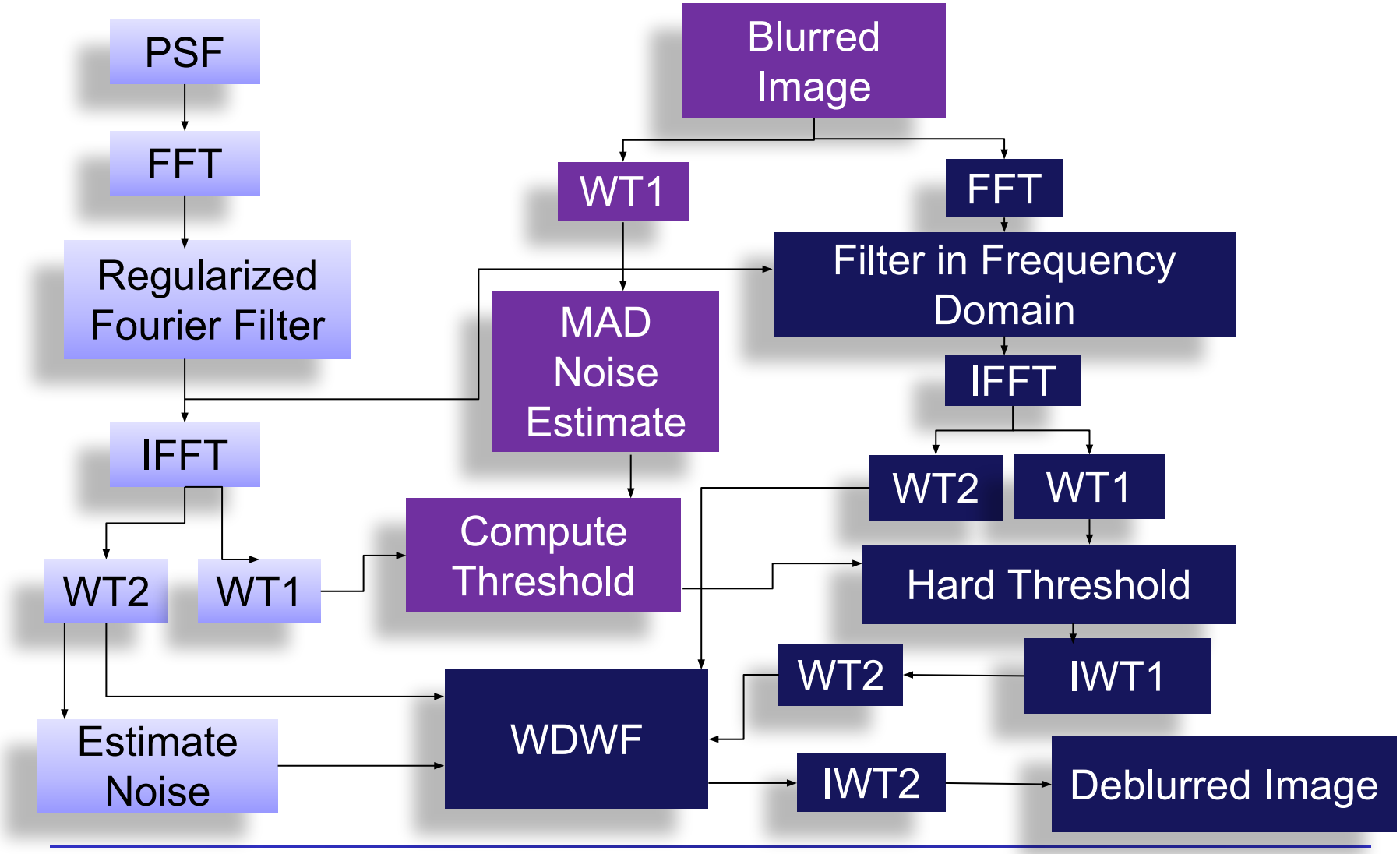


Deblurring

IWT1 - Inverse WT of WT1

WDWF – Wavelet Domain Wiener Filter

ForWaRD Implementation



Results Comparison

Original image

Noisy image, SNR =40dB

Wiener, ISNR =5.61dB, SNR =20.79dB

ForWaRD, ISNR =7.41dB, SNR =22.6dB

Ref[8]: http://dsp.rice.edu/software/forward

ForWaRD Implementation on GPU

- Real time HD video deblurring (30 f/s)
  - Camera + Processor + GPU

- **DDGPU** library (**D**emosaic & **D**eblur on **GPU**)
  - Reduced ForWaRD
    - Remove WDWF, Subframes for Thresholding

- Performance Improvements
  - Reduced memory transfers
  - Reduced unnecessary data movements

Ref[2]: J. Dysart “Embedded Realtime HD Video Deblurring

ForWaRD Performance Results

- Implement DDGPU on GPU cards

Time/frame (ms)	Color	GPU	Improvements
55	Gray	GTX 580	Removed device transfers
100	Color	GTX 580	Reduced Computations
37.5	Color	GTX 670	Full Reduced Algorithm
23	Color	GTX 780	Upgraded Hardware

ForWaRD Implementation Comparison

ForWaRD	ForWaRD on GPU
Huge computations	Parallel Computations (Multi core)
High Memory bandwidth	Minimized bandwidth
Grayscale images	Colored images

Conclusion

- Deconvolution in any single domain is inadequate
- Forward combines fourier & wavelet domains
- Better Estimate than traditional filters
- Improved MSE Good visual quality
- Space variant applications
- Future: Gpu cards Fpga's

References

[1] R. Neelamani, H. Choi, and R. Baraniuk, 'Forward: Fourierwavelet regularized deconvolution for ill-conditioned systems

[2] J. Dysart, B. Brockman, S. Johnes, F. Bacon "Embedded Realtime HD Video Deblurring

[3] C. Gonzalez, E.Woods, "Digital Signal Processing," ,Pearson,

[4] T. Young,J. Gerbrands,J. van Vliet, "Fundamentals of Image Processing," Delft University of Technology.

[5] Y. Nievergelt, "Wavelets Made Easy," Birkhaeuser, 1999.

[6] J. Kalifa and S. Mallat, "Thresholding estimators for linear inverse problems,'Ann. Statist.,', vol. 31,no. 1, Feb 2003.

[7] S. Mallat, A Wavelet Tour of Signal Processing. New York: Academic, 1998.

[8] http://dsp.rice.edu/software/forward

DANKE

Thank you