

Feature-preserving image denoising with multiresolution filters

Yiqian Wang

Electrical and Computer Engineering Department

You-Yi Jau

Electrical and Computer Engineering Department

Abstract—Denoising is a fundamental procedure in image processing. In this project, we investigate the benefit of multiresolution analysis with filter bank in image denoising problem. After reviewing and implementing different single resolution filters, we extensively studied the multiresolution bilateral filter, and generalized the guided filter to the proposed multiresolution guided filter. We further evaluated their feature-preserving performance by the SIFT feature matching experiment. The multiresolution filters showed improved performance compared to the single resolution filters.

Index Terms—Image denoising, bilateral filter, guided filter, wavelet decomposition

I. INTRODUCTION

DENOISING is a fundamental procedure in image processing. Different types of noise can be captured by a digital camera, such as Gaussian noise, salt-and-pepper noise, shot noise, quantization noise and so on. The properties of the noise in an image depends on many factors, such as sensor properties, exposure time, and ISO.

In order to reduce the noise and obtain better visual quality, the image denoising problem have been widely studied and various algorithms have been developed. Most algorithms is based on the assumption that noise is random. One category of algorithms called *image averaging* is based on the law of large numbers, that takes the average of multiple images of the same scene. This method can obtain higher signal-to-noise ratio when the number of images increases.

However, in many applications, only one image with noise is available. The second category of algorithms called *spatial averaging* is often applied in this scenario. In general, the random noise values are less correlated than the clean image values in neighboring pixels, so noise can be smoothed away by local spatial averaging. When the image intensities vary slowly spatially, pixels in a neighborhood have similar values, and therefore averaging them together will still preserve the image.

There are two challenges in the denoising task. The first is the ability to preserve the edge. The assumption of slow spatial variations does not hold at an edge or a corner, and averaging is likely to blur it. However, edges or corners are good features to track, and the balance between noise reduction and feature-preserving ability of the denoising algorithm can seriously affect subsequent higher-level tasks, such as feature detection and matching. This dilemma motivates designing a feature-preserving image denoising algorithm.

The second challenge is that image noise is not necessarily white and may be spatially correlated. The multiresolution analysis have been proven useful for eliminating noise better than processing on a single resolution, as coarse-grain noise becomes fine-grain at higher decomposition level.

In this project, we will design an image denoising algorithm that can effectively eliminate noise while preserving edges and features, and investigate the how wavelet analysis can benefit this task.

II. LITERATURE REVIEW

A. Single-resolution filters

The median filter is a non-linear digital filtering technique that replaces the pixel value by the median value in a neighborhood. The median filter is widely used for image denoising, because it is fast and particular effective for salt-and-pepper noise. The median filter have also shown better edge-preserving property compared to Gaussian blur at small noise level, but the advantage is less significant at higher noise level. The median filter output $\mathcal{I}_m(\mathbf{x})$ for intensity $\mathcal{I}(\mathbf{x})$ at pixel position $\mathbf{x} = (x, y)$ is defined as

$$\mathcal{I}_m(\mathbf{x}) = \text{median}_{\mathcal{N}_d(\mathbf{x})} \mathcal{I}(\mathbf{p}), \quad (1)$$

where $\mathcal{N}_d(\mathbf{x}) = \{(m, n) | x - \frac{d-1}{2} \leq m \leq x + \frac{d-1}{2}, y - \frac{d-1}{2} \leq n \leq y + \frac{d-1}{2}\}$ denotes a window of size d around point \mathbf{x} .

The bilateral filter [1] is a non-linear filter for denoising that can effectively preserve the edge. It replaces the intensity of a pixel by a weighted sum of the neighboring pixels, and the weights are dependent on Gaussian models for both spatial variation and intensity variation. The bilateral filter output $\mathcal{I}_{bl}(\mathbf{x})$ at position $\mathbf{x} = (x, y)$ is defined as

$$\mathcal{I}_{bl}(\mathbf{x}) = \frac{1}{C} \sum_{\mathbf{p} \in \mathcal{N}_d(\mathbf{x})} e^{-\frac{\|\mathbf{x}-\mathbf{p}\|^2}{2\sigma_s^2}} e^{-\frac{(\mathcal{I}(\mathbf{x})-\mathcal{I}(\mathbf{p}))^2}{2\sigma_c^2}} \mathcal{I}(\mathbf{p}), \quad (2)$$

where C is a normalization factor

$$C = \sum_{\mathbf{p} \in \mathcal{N}_d(\mathbf{x})} e^{-\frac{\|\mathbf{x}-\mathbf{p}\|^2}{2\sigma_s^2}} e^{-\frac{(\mathcal{I}(\mathbf{x})-\mathcal{I}(\mathbf{p}))^2}{2\sigma_c^2}}. \quad (3)$$

As shown in the formula, there are three parameters for a bilateral filter, namely the window size d , the standard deviation for intensity σ_c , and the standard deviation for space σ_s .

The guided filter uses another image as guidance to model the spatial variation. The guided filter assumes a local linear model between the guidance and the filtered output, which ensures an edge in the guidance is also in the filtered output.

Denote the guided filter output $\mathcal{I}_g(\mathbf{x})$ at position $\mathbf{x} = (x, y)$ with guidance \mathcal{G} by

$$\mathcal{I}_g(\mathbf{x}) = \frac{1}{|\mathcal{N}_d|} \sum_{\mathbf{p} \in \mathcal{N}_d(\mathbf{x})} W(\mathbf{x}, \mathbf{p}; \mathcal{G}) \mathcal{I}(\mathbf{p}), \quad (4)$$

Then

$$W(\mathbf{x}, \mathbf{p}; \mathcal{G}) = \frac{1}{|\mathcal{N}_d|} \sum_{\mathbf{k}: \mathbf{x}, \mathbf{p} \in \mathcal{N}_d(\mathbf{k})} \left(1 + \frac{(\mathcal{G}(\mathbf{x}) - \mu_{\mathbf{k}})(\mathcal{G}(\mathbf{p}) - \mu_{\mathbf{k}})}{\sigma_{\mathbf{k}}^2 + \epsilon}\right) \quad (5)$$

where $\mu_{\mathbf{k}}$ and $\sigma_{\mathbf{k}}^2$ denotes the mean and variance of guidance \mathcal{G} in the neighborhood $\mathcal{N}_d(\mathbf{k})$. The ϵ parameter in the guided filter is comparable to the variance of intensity in the bilateral filter, i.e. σ_c^2 .

B. Wavelet thresholding

The wavelet decomposition transforms the image into a domain where it can be sparsely represented. In the sparse representation, the signal have a few non-zero supports, and the random noise can usually be represented by small wavelet coefficients close to zero. Thresholding the coefficients can eliminate the noise, but large threshold values also wipes out the texture and details, so the critical problem is the selection of the threshold value.

VisuShrink [2] uses a single threshold for all highpass wavelet coefficients, which is designed to remove Gaussian noise with high probability, but the universal threshold often results in visually over-smooth images.

BayesShrink [3] is an computes a data adaptive thresholds for each wavelet subband using a Bayesian noise estimation method. It adopts a soft-thresholding technique, where the thresholding output for threshold T and input x is

$$g(x; T) = \max(x - T, 0) \quad (6)$$

C. Multiresolution filter

The multiresolution bilateral filter [4] was proposed to better eliminate spatially correlated noise while preserving the edges. The structure of the multiresolution bilateral filter is shown in Fig. 1. A image is first decomposed into its wavelet subbands using perfect reconstruction filters. Then bilateral filtering is applied to the lowpass subbands, and BayesShrink wavelet thresholding is applied to the highpass subbands before reconstructing the image.

Compared with the baseline single-resolution bilateral filter, the multiresolution bilateral filter can potentially eliminate spatially correlated noise components, as the coarse-grain noise becomes fine-grain at higher decomposition level. In the paper, the authors selected 'db8' filters and 4 level decomposition for denoising, but didn't provide justifications for the choice.

D. Learning based methods

The denoising convolutional neural network (DnCNN) [5] is a learning based method that is trained on additive Gaussian noise, and the DnCNN can perform Gaussian denoising with unknown noise level [6]. However, the DnCNN only performs convolution at the same scale. The UNet [6] introduced an CNN architecture that include downsampling and upsampling blocks, which enables the network to perform multiresolution analysis.

III. PROPOSED METHOD

A. Improved multiresolution bilateral filter

In order to better eliminate spatial correlated noise, while preserving the edges and other useful features, we consider the architecture of the multiresolution bilateral filter, and will provide a analysis on the effect of choosing different wavelet and different levels.

The original multiresolution bilateral filter uses the BayesShrink method to threshold highpass coefficients at each level. In order to verify the effect of wavelet thresholding in the multiresolution bilateral filter, we also tested the performance without wavelet thresholding on the highpass branch.

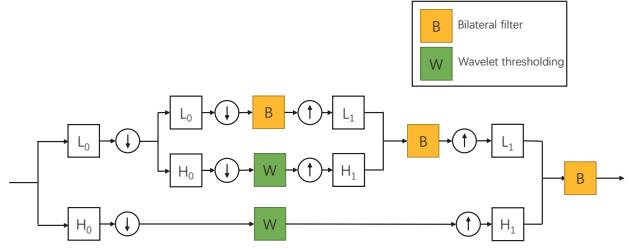


Fig. 1. Structure of a two-level multiresolution bilateral filter [4]

B. Noise estimation

In the bilateral filter, the optimal standard deviation for intensity value σ_c is dependent on the noise level of input image. In [4], it has been shown that the optimal σ_c for bilateral filter is approximately 2 times the noise standard deviation σ . So in this project, we use the robust median noise estimator [3] to estimate the noise variance at each decomposition level, and select $\sigma_c = 2\sigma$ for the bilateral filters at each level.

C. Multiresolution guided filter

We also propose the multiresolution guided filter, that generalizes the guided filter with wavelet decomposition, in the same way as the multiresolution bilateral filter generalizes the bilateral filter. As explained in section II, we use $\epsilon = 4\sigma^2$ for the guided filters at each level.

IV. EXPERIMENTS

In this project, we investigate both the denoising performance and feature preserving performance of different filtering algorithms and their multiresolution counterparts.

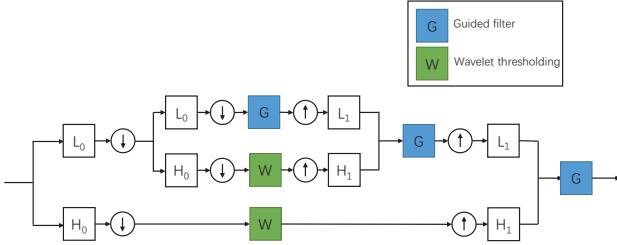


Fig. 2. Structure of a two-level multiresolution guided filter

A. Smartphone image denoising

In the past decade, more and more people tend to take pictures with smartphone cameras instead of taking pictures with DSLR or point-and-shoot cameras. However, the noise level of most smartphone cameras are notably higher than their DSLR counterparts, due to various constraints on mobile devices. Therefore, in this experiment, we will test the denoising performance of the filters using real images captured by smartphones.

Fig. 3 shows the noise in LL component when decomposing a image patch at different level. We can see the noise are spatially correlated in the original scale, shown by similar noise values in neighboring pixels. As the image patch is decomposed to the second level, the noise becomes more fine-grain than at the original scale. However, at higher scale, the bordering artifacts become more prominent. As the image resolution decreases exponentially with number of level, filtering at 3rd or higher level is also more likely to blur the edge. We can also observe that the noise is actually dependent on the image content, as the edge is noticeable even in the noise image, and there appear to be more green color noise in the red pepper.

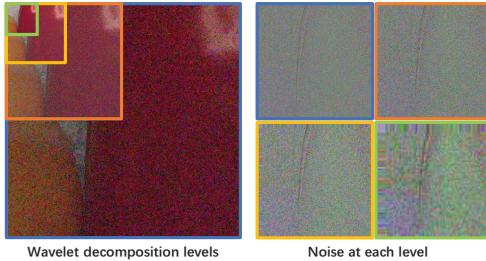


Fig. 3. Noise at different wavelet decomposition level for an example image patch from SIDD [7]. Please zoom in to see details.

We evaluate the image denoising performance of different spatial filters on the small version of the Smartphone Image Denoising Dataset (SIDD) [7]. There are a total of 160 pairs of noisy and clean sRGB image captured by different smartphones. The groundtruth clean image is obtained by averaging 150 images shot at the same position under the same setting.

The denoising performance is evaluated by the mean square error (MSE), peak signal-to-noise ratio (PSNR), and structural

similarity (SSIM) index between the denoised image and the ground truth. The MSE is defined as

$$\text{MSE}(\mathcal{I}_1, \mathcal{I}_2) = \text{Mean}[(\mathcal{I}_1 - \mathcal{I}_2)^2]. \quad (7)$$

The PSNR is defined as

$$\text{PSNR}(\mathcal{I}_1, \mathcal{I}_2) = 10 \cdot \log_{10} \left(\frac{L^2}{\text{MSE}(\mathcal{I}_1, \mathcal{I}_2)} \right), \quad (8)$$

where L denotes the maximum value in the image, which is 255 for 8 bit images in our case. The SSIM is defined as

$$\text{SSIM}(\mathcal{I}_1, \mathcal{I}_2) = \frac{(2\mu_1\mu_2 + c_1)(2\sigma_{12} + c_2)}{(\mu_1^2 + \mu_2^2 + c_1)(\sigma_1^2 + \sigma_2^2 + c_2)}, \quad (9)$$

with μ_j being the mean value of \mathcal{I}_j ($j = 1, 2$), σ_j^2 being the variance of \mathcal{I}_j , σ_{12}^2 being the covariance of \mathcal{I}_1 and \mathcal{I}_2 . $c_1 = (k_1 L)^2$, $c_2 = (k_2 L)^2$ are two stabilizing variables, and $k_1 = 0.01$, $k_2 = 0.03$ by default. As SSIM is only designed for single channel image, we first convert the sRGB images into grayscale for SSIM calculation.

We first randomly selected 40 images from SIDD small to tune the filter parameters. The PSNR, SSIM and MSE at different decomposition level using multiresolution bilateral filter is shown in Table. II. Among the filter types we tested, the 'db4' filter at level 2 achieved the best result. After choosing the filter type, we used window size $d = 11$ and spatial standard deviation $\sigma_s = 1.8$ found by grid search.

The quantitative results for denoising using different filters are shown in Table. I. The wavelet thresholding have very limited improvement compared to the input image. The bilateral filter have better noise reducing performance, but the guided filter preserve better structural similarity. For multiresolution filters, adding the wavelet thresholding in the highpass branch improves the performance. The multiresolution bilateral filter with thresholding achieved the best PSNR and MSE, while the multiresolution guided filter preserves better structure shown by the highest SSIM.

We also included the learning based methods DnCNN and UNet_D from the public benchmark dataset. As the SIDD benchmark dataset is harder than the SIDD small we are using, the DnCNN trained with Gaussian noise shows worse performance than our direct input. But the UNet_D that includes multiresolution analysis performs much better.

Models	PSNR	SSIM	MSE
Input	27.65	0.623	54.39
BayesShrink [3]	29.61	0.750	44.05
Median [8]	32.32	0.845	25.24
Bilateral [1]	33.22	0.841	28.86
Guided [9]	31.75	0.870	32.80
MultiBilateral (no thres.)	34.45	0.843	23.77
MultiBilateral [4]	35.46	0.869	21.64
MultiGuided (no thres.)	33.29	0.809	37.30
MultiGuided	34.86	0.881	22.57
DnCNN [5]	23.66	0.583	–
UNet_D [6]	38.88	0.952	–

Table I. Denoising performance of different filters on SIDD small [7]. The results of DnCNN and UNet_D are from the SIDD benchmark.

Level	Metric	haar	bior3.1	bior3.3	bior3.5	Wavelet type					
						db2	db4	db8	sym2	sym4	sym8
1	PSNR	35.67	36.10	36.15	36.16	35.68	35.68	35.70	35.68	35.72	35.73
	SSIM	0.864	0.869	0.874	0.875	0.864	0.866	0.866	0.864	0.865	0.866
	MSE	25.33	23.30	22.96	22.90	25.22	25.12	25.03	25.22	25.02	24.95
2	PSNR	36.06	36.27	35.73	35.42	35.20	37.02	36.78	36.34	36.55	35.98
	SSIM	0.900	0.870	0.866	0.862	0.817	0.904	0.875	0.870	0.892	0.823
	MSE	22.10	22.97	23.62	26.03	29.57	19.00	20.80	22.97	20.58	27.63
3	PSNR	36.81	34.46	35.51	35.67	36.91	36.94	36.96	36.91	36.96	36.98
	SSIM	0.861	0.824	0.845	0.848	0.862	0.863	0.863	0.862	0.863	0.863
	MSE	22.15	31.43	25.87	25.14	21.62	21.46	21.35	21.62	21.35	21.23
4	PSNR	35.03	31.86	33.24	33.47	35.10	35.08	35.09	35.10	35.10	35.12
	SSIM	0.795	0.748	0.783	0.787	0.797	0.798	0.798	0.797	0.799	0.800
	MSE	31.33	48.09	38.14	36.83	30.70	30.67	30.58	30.70	30.51	30.42

Table II. **Parameters tuning for multiresolution bilateral filter:** effect of different wavelets types and different decomposition level. The PSNE, SSIM, MSE are tested on 40 randomly selected samples from SIDD small [7]. The 'db4' filter at 2 level performs the best.

The qualitative results are shown in Fig. 4. We can see the wavelet thresholding alone cannot effectively eliminate the noise. The boundary of highlights are blurred in the images by median filter, but preserved by the other four filter. The single-resolution filters alone cannot fully remove the noise shown by small colorful blocks, but the multiresolution filters have better denoising result.

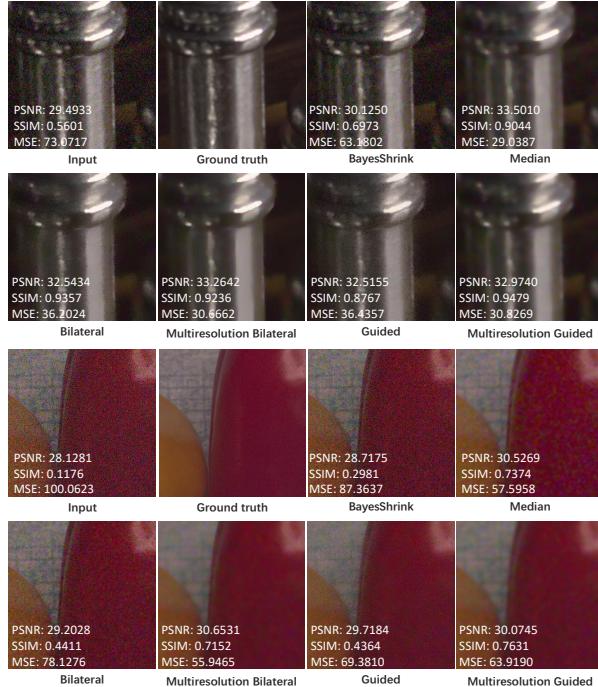


Fig. 4. Denoising performance of different filters for two example images on SIDD small [7]

B. Feature matching

We further evaluate the performance of filters under feature matching task, including keypoint detection and description metrics. We use the metrics described in [10] for evaluating keypoint detection and description. HPatches [11] dataset is used in the task. The dataset contains 116 scenes with 57

	Homography Estimation			Detector Metrics	Descriptor Metrics			
	$\epsilon = 1$	$\epsilon = 3$	$\epsilon = 5$		Rep.	MLE	NN mAP	M. Score
<i>None</i>	0.137	0.176	0.196	0.314	1.75	0.417	0.138	
<i>Median</i>	0.0784	0.118	0.137	0.403	1.83	0.359	0.101	
<i>Bilateral</i>	0.137	0.157	0.196	0.36	1.8	0.415	0.124	
<i>MultiBilat. (no th.)</i>	0.157	0.255	0.275	0.284	1.74	0.448	0.173	
<i>MultiBilat.</i>	0.176	0.235	0.255	0.307	1.74	0.449	0.178	
<i>MultiGuided</i>	0.098	0.118	0.176	0.498	1.77	0.328	0.0903	

Table III. **HPatches Homography Estimation for different filters.** Different filters are plugged into the homography estimation pipeline to reduce noise. Noise level is set to 10 and assumed to be known.

Multi-resolution bilateral filters perform better than others.

	Sigma	Homography Estimation			Detector Metrics	Descriptor Metrics			
		$\epsilon = 1$	$\epsilon = 3$	$\epsilon = 5$		Rep.	MLE	NN mAP	M. Score
<i>None</i>	5	0.235	0.255	0.275	0.35	1.71	0.514	0.185	
	10	0.137	0.176	0.196	0.314	1.75	0.417	0.138	
	15	0.118	0.196	0.216	0.287	1.78	0.38	0.11	
	20	0.0784	0.118	0.137	0.261	1.8	0.324	0.0844	
	25	0.0784	0.118	0.157	0.242	1.82	0.263	0.0616	
<i>MultiBilat.</i>	5	0.255	0.294	0.294	0.351	1.7	0.519	0.206	
	10	0.176	0.235	0.255	0.307	1.74	0.449	0.178	
	15	0.157	0.176	0.196	0.266	1.77	0.394	0.155	
	20	0.157	0.176	0.196	0.233	1.8	0.355	0.137	
	25	0.137	0.157	0.157	0.206	1.79	0.312	0.125	

Table IV. **HPatches Homography Estimation for different noise levels.** Multi-resolution bilateral filter with thresholding is applied to compare with baseline under different noise levels, from 5 to 25. As noise increases, the performance drops for both methods. However, our method achieve better performance. Numbers are **highlighted** by comparing the two filters under noise level 5 and 25.

scenes in illumination changes and 59 scenes in viewpoint changes. There are in total of 696 images, which forms 580 pairs with ground truth homography matrix. The image is cropped into 480×640 .

The evaluation pipeline is described in 5, where SIFT [12] feature extractor is used across all tasks. With the input image from the dataset, we apply Gaussian additive noise to the image. Then, a filter is applied to recover the corrupted image. The recovered image is further fed into SIFT feature extractor to find sparse feature points and corre-

Model References		Settings			Parameters		
Catogories	Symbols	Sift (Si)	Multi-resolution	Thresholding	Dist.	Sigma color	Sigma space
Baseline	None	✓			-	-	-
	Median	✓			5	-	-
	Bilateral	✓			11	$2 \times \text{sigma}$	1.8
	MultiBilateral (no thres.)	✓	✓		11	$2 \times \text{sigma}$	1.8
Our methods	MultiBilateral	✓	✓	✓	11	$2 \times \text{sigma}$	1.8
	MultiGuided	✓	✓		1	$2 \times \text{sigma}$	-

Table V. **The reference table for filters in feature matching.** Sift is used for all methods. Parameters are listed in the table. Bilateral filters follow the paramters in [4]. Median and guided filter are decided visually from one image in the dataset.

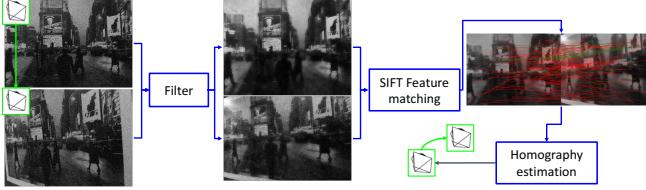


Fig. 5. **Pipeline for evaluation under feature matching.** A pair of images are fed into the pipeline to predict homography matrix. The filters are applied before SIFT feature matching.

pondences. Correspondences are used to compute homography matrix between the pair of images using *OpenCV* function (`findHomography()` with RANSAC). We compare the performance of different filters under one noise level. The filters including median filter, bilateral filter(Bilateral), multi-resolution bilateral filter(MultiBilateral no thres.), and multi-resolution bilateral filter with thresholding (MultiBilateral), multi-resolution guided filter with thresholding (MultiGuided). The parameters are as Tab. V. The noise level, sigma, is set to be 10. Also, we evaluate our proposed filter under different noise levels, from 5 to 25. The result shows that by filtering the images, we can boost the performance of feature matching. Also, our proposed filter works marginally better than existing multi-resolution bilateral filters.

Homography estimation is performed to evaluate both the detector and the descriptor through recovering homography matrix between a pair of images. The metric measures the inlier ratio based on the distance of four corners against that from ground truth. Assuming four corners of one image is defined as $\mathbf{c}_1, \mathbf{c}_2, \mathbf{c}_3, \mathbf{c}_4$, we can project the corners using ground truth homography matrix into $\mathbf{c}'_1, \mathbf{c}'_2, \mathbf{c}'_3, \mathbf{c}'_4$. Also, the four corners projected by estimated homography are $\hat{\mathbf{c}}'_1, \hat{\mathbf{c}}'_2, \hat{\mathbf{c}}'_3, \hat{\mathbf{c}}'_4$. We define

$$\text{CorrH} = \frac{1}{N} \sum_{i=1}^N \left(\left(\frac{1}{4} \sum_{j=1}^4 \|\mathbf{c}'_{ij} - \hat{\mathbf{c}}'_{ij}\| \right) \leq \varepsilon \right), \quad (10)$$

where CorrH measures the inlier ration that the mean distance again ground truth corners lies under a threshold ε . We follow the setting from [10] with $\varepsilon = 1, 3, 5$ (pixels).

Detector is evaluated using repeatability and mean localization error (MLE). Repeatability is used to observe how much percentage of the detected keypoints on one image is detected

on the other within a threshold ε . We define $\{\hat{\mathbf{x}}_1, \dots, \hat{\mathbf{x}}_K\}$ as ground truth corners projected from keypoints on the first image. The correctness of a keypoint on the second image is defined as

$$\text{Corr}(\mathbf{x}) = (\min_j \|\mathbf{x} - \hat{\mathbf{x}}_j\|) \leq \varepsilon. \quad (11)$$

Accumulating all points, we can measure the keypoint repeatability across a pair of images. In the evaluation, we set $\varepsilon = 3$ (pixels) and mask out the non-overlapping area between two images.

For MLE, we measure the localization error as defined in [10],

$$\text{LE} = \frac{1}{N} \sum_{i: \text{Corr}(\mathbf{x}_i)} \min_{j \in \{1, \dots, K\}} \|\mathbf{x}_i - \hat{\mathbf{x}}_j\|. \quad (12)$$

Localization error is only on the repeated keypoint set Corr . The error lies between 0 and ε , the lower the better.

Descriptor is evaluated using nearest neighbor mean average precision (NN mAP) and matching scores (M. score). Correspondences are found from keypoint detectors and descriptors through kNN search with ratio test. The distance between two descriptors are recorded to compute average precision. Inlier correspondences are identified in RANSAC for homography estimation. Matching score (M. score) is a metric for both detection and description. It measures the ratio of inlier correspondences and all keypoints in the pair of images in the shared area.

$$\text{M. score} = \frac{\# \text{ of Inlier correspondences}}{\# \text{ of points in shared region}} \quad (13)$$

We compare different filters under the metrics mentioned above in Tab. III. The *MultiBilateral* works the best while *median* and *MultiGuided* work worse than the baseline. The evaluation of guided filter shows that good performance in classic metrics (MSE, PSNR) may not lead to good performance in high-level task. We furtre evaluate the *MultiBilateral* under different noise levels, shown in Tab. IV. Our approach shows more stable performance across different noise levels, whereas the baseline method performs sharply worse when the noise level increases. We show the potential benefits of adding filters as preprocessing step for high-level tasks.

Besides quantitative results, qualitative results are shown in Fig. 6. We can observe higher percentage of inlier correspondences are found after filtering using our *MultiBilateral* filter.

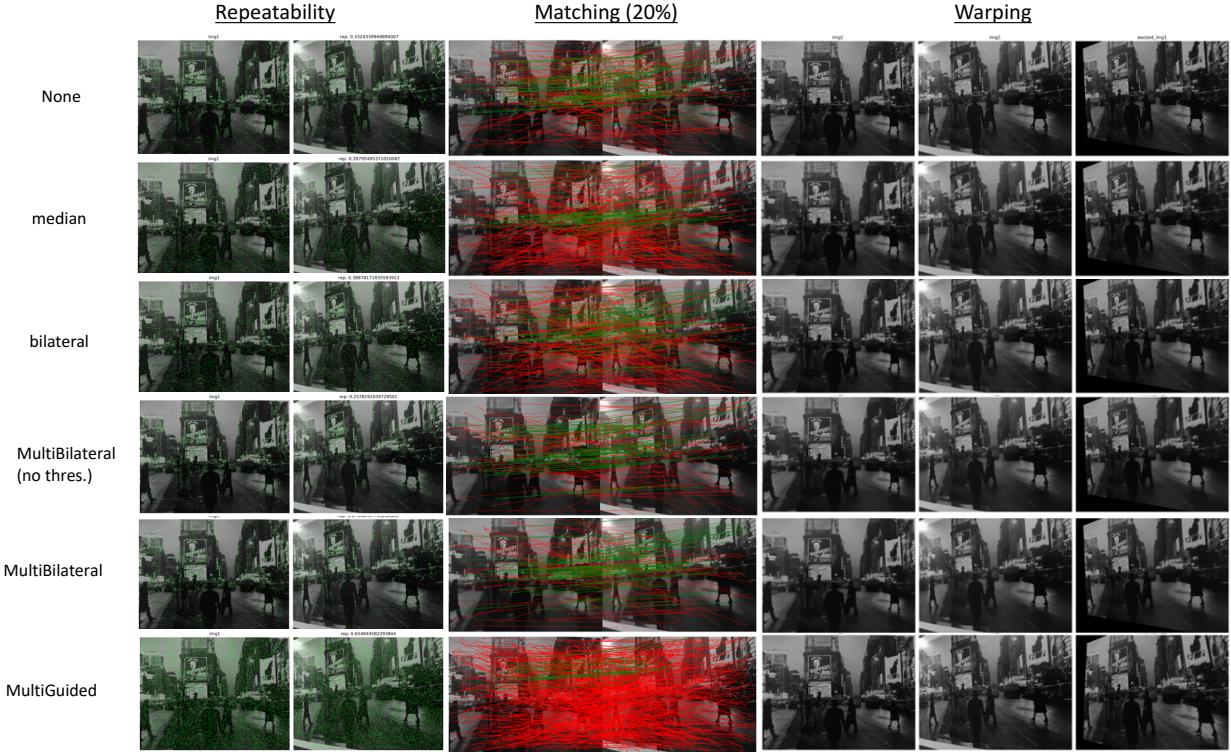


Fig. 6. Qualitative results for different filters. Different filters are shown in each row, where visualization is done under the same image pair. Repeatability is shown in the first column. Matching with inliers (green lines) and outliers (red lines) are shown in the second column. The filtered image pair and warped image is shown in the 3rd column. As we can see, although MultiGuided has the most dense keypoint detection, *MultiBilateral* has the cleanest matching. We only plot randomly sampled 20% of total correspondences.

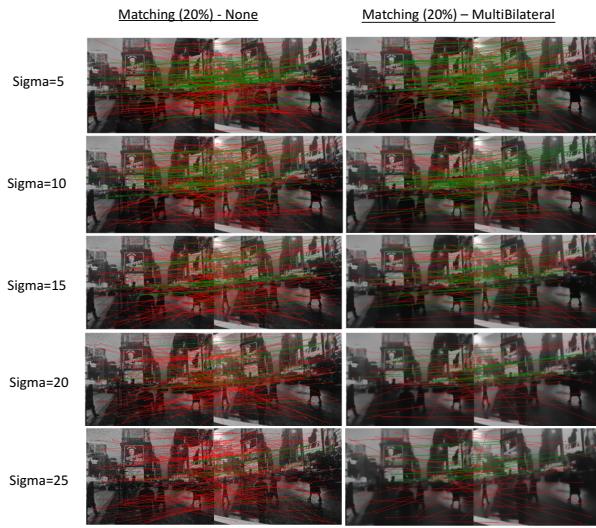


Fig. 7. Qualitative results for filters under different noise levels. The filter *MultiBilateral* is compared with the pipeline without filter, under noise levels from 5 to 25. Our *MultiBilateral* predicts cleaner matching than that with no filter.

Repeatability and warping results are also shown in Fig. 6. When testing under different noise levels, matching is shown in Fig. 7. Matching found by our *MultiBilateral* filter is cleaner than that with no filters. The cleaner matching usually leads to

better homography estimation.

V. CONCLUSION

In this project, we reviewed and implemented several types of single-resolution filters for image denoising, and investigated the benefits of combining them with wavelet analysis. We extensively studied the multiresolution bilateral filter, and proposed the multiresolution guided filters. We tested their denoising performance using standard metrics including MSE, PSNR, SSIM, and further tested their feature-preserving performance by the SIFT feature matching experiment. The results showed that multiresolution analysis can remove noise effectively while preserving the features.

REFERENCES

- [1] C. Tomasi and R. Manduchi, “Bilateral filtering for gray and color images,” in *Sixth International Conference on Computer Vision (IEEE Cat. No.98CH36271)*, Jan. 1998, pp. 839–846, ISSN: null. [1](#), [3](#)
- [2] D. L. Donoho and J. M. Johnstone, “Ideal spatial adaptation by wavelet shrinkage,” *biometrika*, vol. 81, no. 3, pp. 425–455, 1994. [2](#)
- [3] S. G. Chang, B. Yu, and M. Vetterli, “Adaptive wavelet thresholding for image denoising and compression,” *IEEE transactions on image processing*, vol. 9, no. 9, pp. 1532–1546, 2000. [2](#), [3](#)
- [4] M. Zhang and B. K. Gunturk, “Multiresolution Bilateral Filtering for Image Denoising,” *IEEE Transactions on Image Processing*, vol. 17, no. 12, pp. 2324–2333, Dec. 2008. [2](#), [3](#), [5](#)
- [5] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang, “Beyond a Gaussian Denoiser: Residual Learning of Deep CNN for Image Denoising,” *IEEE Transactions on Image Processing*, vol. 26, no. 7, pp. 3142–3155, July 2017. [2](#), [3](#)

- [6] O. Ronneberger, P. Fischer, and T. Brox, “U-net: Convolutional networks for biomedical image segmentation,” in *International Conference on Medical image computing and computer-assisted intervention*. Springer, 2015, pp. 234–241. [2](#), [3](#)
- [7] A. Abdelhamed, S. Lin, and M. S. Brown, “A high-quality denoising dataset for smartphone cameras,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. [3](#), [4](#)
- [8] T. Huang, G. Yang, and G. Tang, “A fast two-dimensional median filtering algorithm,” *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 27, no. 1, pp. 13–18, 1979. [3](#)
- [9] K. He, J. Sun, and X. Tang, “Guided Image Filtering,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 6, pp. 1397–1409, June 2013. [3](#)
- [10] D. DeTone, T. Malisiewicz, and A. Rabinovich, “SuperPoint: Self-Supervised Interest Point Detection and Description,” *arXiv:1712.07629 [cs]*, Dec. 2017, arXiv: 1712.07629. [4](#), [5](#)
- [11] V. Balntas, K. Lenc, A. Vedaldi, and K. Mikolajczyk, “HPatches: A benchmark and evaluation of handcrafted and learned local descriptors,” *arXiv:1704.05939 [cs]*, Apr. 2017, arXiv: 1704.05939. [4](#)
- [12] D. G. Lowe et al., “Object recognition from local scale-invariant features,” in *iccv*, 1999, vol. 99, pp. 1150–1157. [4](#)