Image Smoothing Methods for Structure Preservation in Computer Vision

Manar Abdulkareem Al-Abaji and Mohammed Hazim Alkawaz
Department of Computer Sciences, College of Education for Pure Science, University of Mosul, 41002 Nineveh, Iraq
manar_alabaji@uomosul.edu.iq, mohammed.ameen@uomosul.edu.iq

Keywords: Edge Detection, Fractional Riesz Derivative, Structure Preserving, Image Smooth, Multi-Channel Smoothing.

Abstract: Image smoothing is a fundamental pre-processing step in multimedia and computer vision applications, as it suppresses texture and high-frequency details to facilitate subsequent processing and analysis. A smoothed image is essential for many image processing tasks, such as sharpening, edge detection, abstraction, and detail enhancement. However, a critical challenge in image smoothing lies in balancing the attenuation of high-frequency details with the preservation of prominent structural information. Many existing smoothing methods struggle to achieve this balance, often resulting in excessive blurring, loss of important details, or the introduction of undesired artifacts. To address these limitations, this paper proposes a novel image smoothing method based on the Fractional Riesz Derivative. The proposed approach is designed to smooth color images while effectively preserving their essential structural features. Its performance assessed with over 330k of diverse color images from COCO data set, compared with other six state-of-the-art smoothing techniques in order to highlight the proposed method's effectiveness. Experimental results show that the proposed method outperforms to other methods through three quantitative evaluation metrics.

1 INTRODUCTION

Image smoothing is fundamental paradigm in the applications of computer vision, it considered as a cornerstone in various post-processing, like ton mapping, image segmentation, image abstraction, details boosting, etc. [1]. Edges-preserving smoothing is a fundamental technique in computer graphics and computer vision. Its main goal is to attenuate image details and at the same time keep the important image's edges and structural features looking good [2]. This technique proved its viability in various areas. For example, in self-driving, it helps the vehicles in recognizing the boundaries of the road accurately. In medical imaging, it highlights the organs shape and the boundaries of the lesion, which facilitate the clinical diagnosis for doctors. While in artistic creativity based on images, it removes the distracting details and maintaining the important structural of the image, this help to producing artworks with powerful visual aesthetics [3], [4].

During the past decades, diverse image smoothing methods was presented that can be classified as weighted averaging methods, optimization-based methods, and deep learning methods. As for the weighted averaging methods, it is also called local filtering. It carry out filtering in local regions and obtains its output by calculating the weighted average of the adjacent pixel values within local window, thus avoidance the excessive smoothing of edges [3]. These methods include filters like Bilateral filter [5], joint bilateral filter [6], guided filter [7], weighted guided filter [8], rolling guidance filter [9], etc. the main drawbacks of these methods, it tends to produce unwanted artifacts such as gradient reversals or halos around edge areas [10]. While the optimization-based methods carry out its process on the image's overall structure through inserting a regularization term to accomplish smoothing process [3]. A common methods are those based on total variation like L_0 [11], RTV [12], SDF [13], ILS [14], G-smooth [15], SSF [16], etc. these types of methods suffers a tradeoff between the cost and the performance of smoothing process. In other words, it consumes high computation to get a good image smooth quality [10]. As for the deep learning methods, it can learn the structural and semantic image features from huge image datasets utilizing deep convolutional layers, which achieves efficient image smoothing with good edges preserving [3]. Some of these methods are: WT-L1 [17], S2DGNet [18], DualCNN [19]. The limitation of these methods is needed a huge and large-scale of paired training data [10].

From this perspective, proposed a method addresses the limitations of the previous methods, so that well smooth the image and preserve its important structures without doing massive computation would be contributed in this aspect. This study introduces a new smoothing method based on the Fractional Riesz Derivative (FRD) concept to smooth color image and achieving a balance between the image smoothing with preserving its essential structure. The motivation behind choosing FRD in image smoothing is that it uses fractional orders, which enable it to accurately identify edges by balancing the preservation of fine details with the reduction of noisy effects in the images. The methodology of this method relay on extracting edges information from the image luminance using FRD, then an adaptive smoothing weight is calculated to avoid the excessive smoothing at edges, after that a normalized smoothing process is applied on each color channel separately (multi-channel smoothing) to preserve the edges and maintains the image brightness mean. The proposed method was tested on various color images and the results compared with other six smoothing methods. Our method gave promising results. which demonstrate its validation in image smoothing. The rest of this study organized as follows: the proposed method is explained minutely in section 2, while the results of experiments and comparison are presented in section 3, finally, section 4 include a brief conclusion.

2 PROPOSED METHOD

The proposed method averts edges blurring (excessive smooth) and color distortion at the edges. This method relay on extracting edges information (high frequency component) from the image luminance using FRD then a normalized smoothing process is applied to reduce this information and at the same time preserve the edges and maintains the contrast of the image. The methodology of the method consists of many stages: the first stage includes extract luminance image/grayscale. Since the structural edges are geometric features and not a color features, processing each color channel (R, G, B) separately can cause color shift. Therefore, the first stage of the proposed method includes calculate the luminance image/grayscale (L) from the input color image (I). while the second stage done by calculate the FRD. This stage includes edges detection using the FRD, the (L) image is converted to the frequency domains to get ($\hat{L}$) then calculate Riesz derivatives in four directions to cover all the geometrical structures. Then inverse Fourier transform is applied to get the spatial edges map. Converted the (L) image to the frequency domains done by the following equation:

$$\hat{L}(u,v) = \mathcal{F}\{L(x,y)\}, \quad (1)$$

where:

- L is luminance image;
- $\mathcal{F}$ is Fourier transform;
- $\hat{L}$ is the image in the frequency domain.

The Riesz kernels (the transfer functions for Riesz in the frequency domain in x and y directions) is as follows [20]:

$$\mathcal{R}_x^{\alpha}(u,v) = (jw_x)^{\alpha}, \quad (2)$$

$$\mathcal{R}_y^{\alpha}(u,v) = (jw_y)^{\alpha}, \quad (3)$$

where:

- $\mathcal{R}_x^{\alpha}$ is the frequency response at the x direction;
- $\mathcal{R}_y^{\alpha}$ is the frequency response at the y direction;
- W_x and W_y are uniform angular frequencies, $j = \sqrt{-1}$;
- α is the fractional derivative order which control the amplification of the structural edges, $0<\alpha<2$.

In this study $\alpha=1.8$ which provide a good balance between smoothing and edges discrimination. Utilizing the linearity steerable of Riesz derivative, the calculation of the derivatives in four directions (θ=0°, 90°, 45°, 135°) can done using the following equation [21]:

$$\widehat{D}_{\theta} = \hat{L} \cdot (cos(\theta)\mathcal{R}_x^{\alpha} + sin(\theta)\mathcal{R}_y^{\alpha}). \quad (4)$$

After that, the inverse Fourier transform is applied to get the spatial edges map using the following equation:

$$D_{\theta}(x,y) = |\mathcal{F}^{-1}\{\widehat{D}_{\theta}\}|. \quad (5)$$

The third stage include calculating adaptive smoothing weights, its main goal is to obtained weight function which is small at the edges to avoid the smoothing, and big in soft area to encourage the smoothing. Perona and Malik present an anisotropic diffusion function which gave small value when the gradient is significant (edges) and great value when the gradient is minute (flat area) using the following equation [22]:

$$g(\nabla I) = \frac{1}{1+\left(\frac{\|\nabla I\|}{k}\right)^2}, \quad (6)$$

where ‖∇I‖ is image gradient magnitude, k is threshold constant. In this paper, (6) is adopted to became more generalized and used it to calculate adaptive smoothing weights for each direction (θ=0°, 90°, 45°, 135°). Replacing the squaring with decay power β>2, and the threshold constant (k) by the mean of the edge response, make the transition between the edge and the flat area became more gradually. This reduces the effect of the staircasing. A suitable value of β=5, this value prevent edges blur occurs when the value of β small at the same time prevent the excessive edges preserving when the value of β high. The adopted equation becomes as follow:

$$w_\theta(x,y) = \frac{1}{1+\left(\frac{D_\theta(x,y)}{h}\right)^{\beta}}, \quad (7)$$

$$h = \mu(D_\theta), \quad (8)$$

where:

- is decay power;
- h is the mean of the edge response;
- D_θ the spatial edges map;
- W_θ is the adaptive smoothing weights.

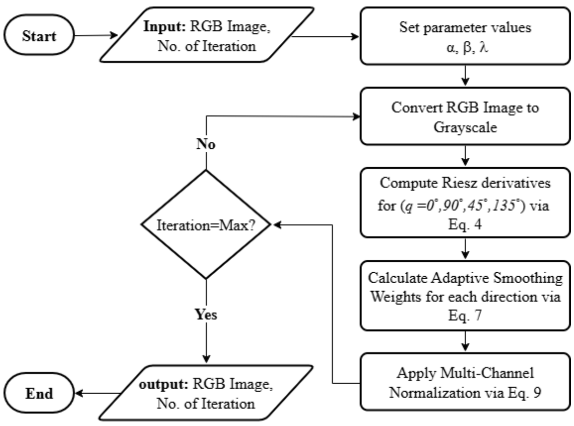

Figure 1: The block diagram of the proposed method.

The fourth stage include applying a normalized smoothing on each color channel separately (multi-channel smoothing). This stage depends on the idea of the of utilizing the normalization function presented in [5], but is adopted to prevent color shift and to ensure the preserving of image mean brightness (prevent darkening issue). The adopted equation becomes as follow:

$$F_c^{(t+1)}(x,y) = \frac{F_c^{(t)}(x,y)+\lambda\cdot\sum_\theta\left(w_\theta(x,y)\cdot N_\theta(x,y)\right)}{1+\lambda\cdot\sum_\theta 2w_\theta(x,y)}, \quad (9)$$

where:

- $F_c^{(t)}(x,y)$ is the image pixel value at channel c;
- c∈(Red, Green, Blue);
- λ is the smooth strength factor;
- N_θ is the sum of neighbor pixels in θ direction.

In this study a suitable value of λ=0.2 which get a good smoothing with edges preserving. The adopted equation joins the current value of the pixel with the weighted neighbors of it then normalized by the total weights. Factor '2' in the denominator takes into account two neighbor pixels in each direction.Figure 1 illustrate the block diagram of the proposed method.

3 RESULTS AND DISCUSSION

This section summarizes the evaluation metrics, data set, the experiments and comparisons results. With respect to the data set utilizing in this study, the COCO data set [23] which is an abbreviation for "Microsoft Common Objects in Context" is used to assess the performance of the proposed method. This data set contains over 330k images with miscellaneous views and sizes. Two hundred images will employment in this study.

The proposed method was compared with six smoothing methods that are: SABF [24], L_0 [11], T_TV [25], RGF [9], SSF [16], RTV [12] and the comparison results was evaluated using three different IQA metrics which are: PSNR, SSIM [26], and FISH [27]. The PSNR and SSIM are full reference metrics that are used for measuring the peak signal to noise ratio and structure similarity of images, respectively. High values of PSNR and SSIM indicate a high image quality. While FISH is a no-reference metric, measuring the image sharpness. High value of FISH indicates the potent image sharpness. Regarding to the runtime, it was recorded for the competing algorithms and for the proposed method to demonstrate the computation complexity. The computer equipment used in this study involves CPU (Intel core i7), RAM (16 GB). MATLAB R2023a is used as a programing language. Figure 2 and Figure 3 present the outcomes of applying the proposed method to set of nature images. Figures 4-8 show the comparison outcomes. Tables 1 to 4 presented the recorded metric values and runtime. Table 1 displays the PSNR values, Table 2 shows the SSIM values, Table 3 presents the FISH values as well as Table 4 illustrate the runtime results.

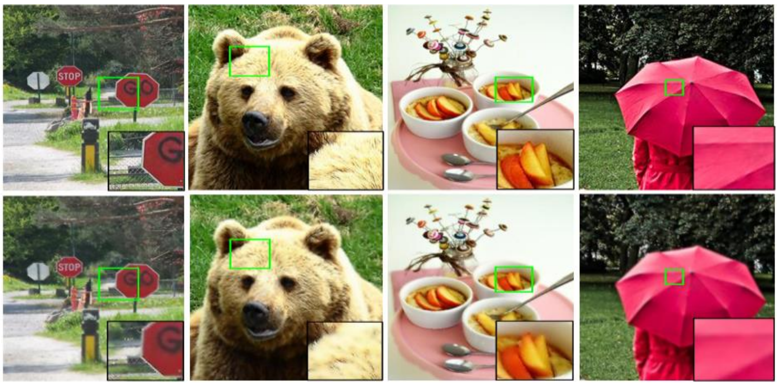

Figure 2: Experimental outcomes of smoothing a set of nature images (Batch 1). The first row are the nature images, the second row are the smoothing images using iteration value (15, 20, 33, 30) respectively.

Figures 9-12 display these metrics and runtime values in the chart form. Figure 9 illustrates the average PSNR values for our proposed method and the six smoothing methods while Figure 10 shows the average SSIM values. The average FISH values are illustrated in Figure 11 and Figure 12 displays the average runtime results for the comparison. As shown in Figure 2 and Figure 3, various images were smoothed using the proposed method. The high-frequency components of the resultant images were noticeably attenuating at the time but still contained the essential structural and preserving image mean brightness without causing color distortion or any artifacts. And thus, the proposed method success in smoothing images with different details and texture, reduce its high frequency components and preserve its vital structural.

Figure 3: Experimental outcomes of smoothing a set of nature images (Batch 2). The first row are the nature images, the second row are the smoothing images using iteration value (17, 20, 36, 31) respectively.

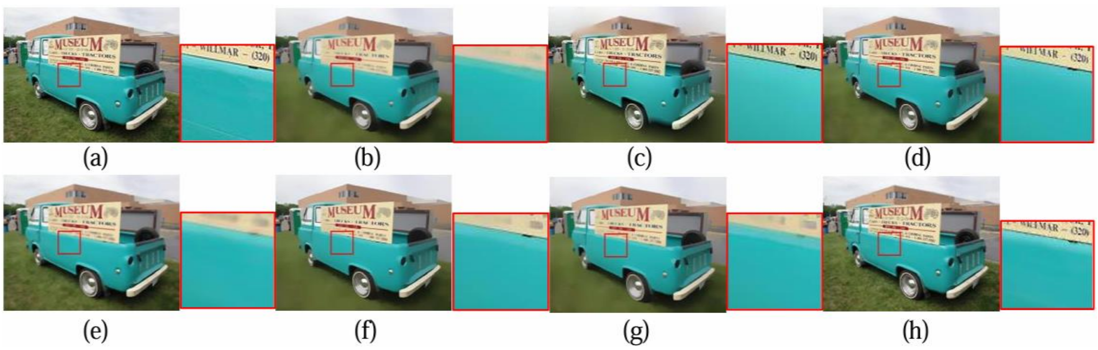

Figure 4: Comparison outcomes (Batch 1): a) Original image, b) SABF, c) L0, d) T_TV, e) RGF, f) SSF, g) RTV, h) the Proposed algorithm.

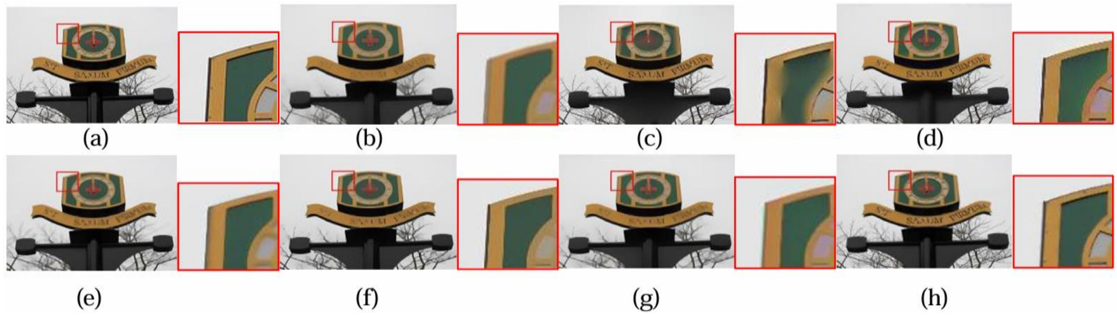

Figure 5: Comparison outcomes (Batch 2): a) Original image, b) SABF, c) L0, d) T_TV, e) RGF, f) SSF, g) RTV, h) the Proposed algorithm.

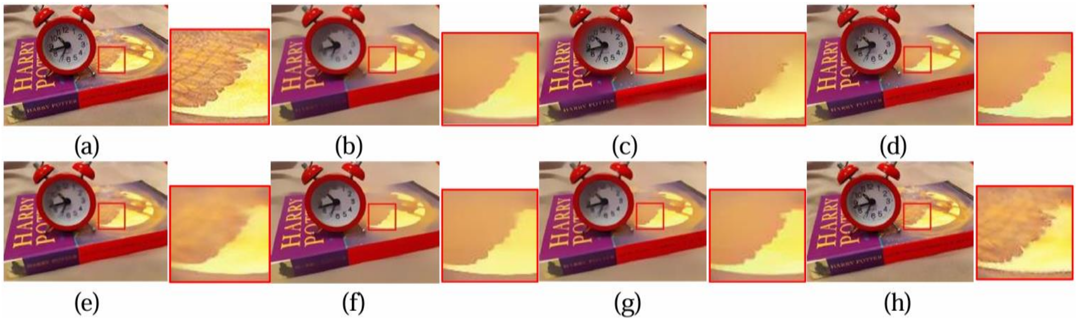

Figure 6: Comparison outcomes (Batch 3): a) Original image, b) SABF, c) L0, d) T_TV, e) RGF, f) SSF, g) RTV, h) the Proposed algorithm.

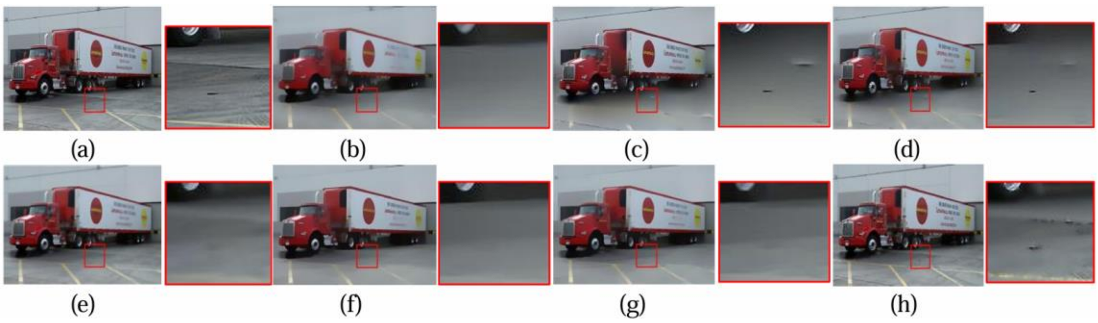

Figure 7: Comparison outcomes (Batch 4): a) Original image, b) SABF, c) L0, d) T_TV, e) RGF, f) SSF, g) RTV, h) the Proposed algorithm.

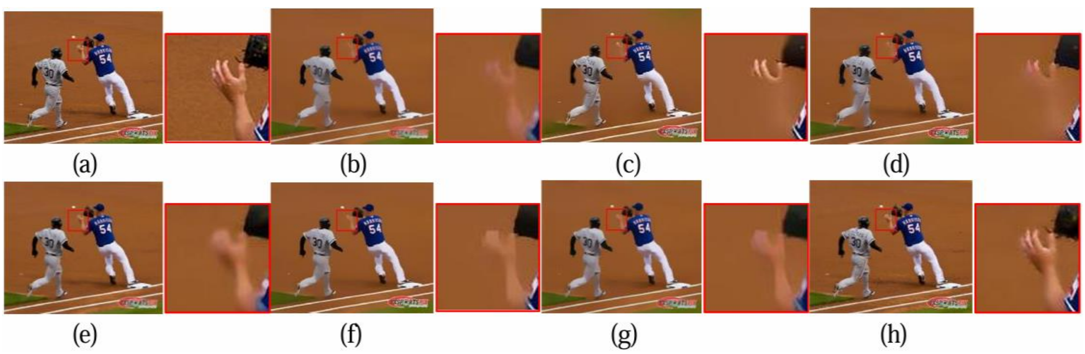

Figure 8: Comparison outcomes (Batch 5): a) Original image, b) SABF, c) L0, d) T_TV, e) RGF, f) SSF, g) RTV, h) the Proposed algorithm.

Table 1: The recorded PSNR ↑ scores for comparison.

Methods	Figure 4	Figure 5	Figure 6	Figure 7	Figure 8	Average
SABF [24]	21.9932	18.3579	22.7764	20.0906	19.9554	20.6347
L0 [11]	25.6077	20.3579	22.7764	23.0906	23.9554	23.1576
T_TV [25]	22.8650	21.7636	24.0728	24.6751	25.4780	23.7709
RGF [9]	21.0307	21.3073	23.8143	22.2469	22.3619	22.1522
SSF [16]	24.9304	28.1219	24.0107	25.3210	27.9004	26.0569
RTV [12]	22.3376	24.8249	22.7698	22.7812	23.8932	23.3213
Proposed	28.3600	27.4900	28.4500	30.1400	31.8600	**29.2600**

Table 2: The recorded SSIM ↑ scores for comparison.

Methods	Figure 4	Figure 5	Figure 6	Figure 7	Figure 8	Average
SABF [24]	0.8011	0.7646	0.8502	0.7222	0.8227	0.7922
L0 [11]	0.8017	0.8258	0.8263	0.7074	0.9039	0.8130
T_TV [25]	0.8932	0.8612	0.9043	0.8790	0.9380	0.8951
RGF [9]	0.8361	0.8179	0.9008	0.7744	0.8936	0.8446
SSF [16]	0.8066	0.8294	0.9126	0.7709	0.9118	0.8463
RTV [12]	0.8846	0.8751	0.7632	0.8945	0.9391	0.8713
Proposed	0.9335	0.9373	0.9580	0.9090	0.9813	0.9438

Table 3: The recorded FISH ↑ scores for comparison.

Methods	Figure 4	Figure 5	Figure 6	Figure 7	Figure 8	Average
SABF [24]	5.3268	5.8583	6.1946	3.8593	4.5846	5.1647
L0 [11]	9.8443	10.5709	8.6467	10.9120	12.2211	10.4390
T_TV [25]	9.8770	13.0232	11.8905	10.3852	9.3374	10.9027
RGF [9]	8.8979	10.6038	10.1965	6.3114	7.7352	8.7490
SSF [16]	8.0777	12.0061	9.3042	10.2770	9.8792	9.9088
RTV [12]	7.7179	7.2386	8.6192	6.5272	9.5947	7.9395
Proposed	14.1584	17.5441	13.2778	11.6063	12.0745	**13.7322**

Table 4: The recorded Runtime ↓ for comparison.

Methods	Figure 4	Figure 5	Figure 6	Figure 7	Figure 8	Average
SABF [24]	10.1775	9.4170	4.1738	6.4571	8.9352	7.8321
L0 [11]	47.3447	33.5022	20.6619	34.7982	37.5474	34.7709
T_TV [25]	1.3616	1.3391	0.9621	1.3543	1.4871	**1.3008**
RGF [9]	36.2463	35.6756	22.2250	32.4565	39.8815	33.2970
SSF [16]	10.2992	10.2361	7.3516	11.0153	11.2709	10.0346
RTV [12]	1.7950	1.7417	1.1425	1.5445	1.9975	**1.6442**
Proposed	2.6807	3.2267	1.8778	2.3498	3.5492	**2.7368**

Figures 4 to 8 and Tables 1 to 4 offer the outcomes of the comparison between the proposed method and six smoothing methods on natural images. As notes that different outcomes were obtained when applying the methods because they depend on different concepts. Therefore, the performance of methods was recorded. The input image has wealthy structural information in addition to high frequency component like details, edges, and text regions. These information and component are very important for the human visual perceptual. When utilizing the SABF method in smoothing the input image, it tends to over smooth the image by removing not only the high frequency component but also the important structural information which lead to poor visual quality. This explains why it gain the lowest scores in all metrics also it has a moderate execution time. Although L_0 method preserve some of image's edges and obtains above-average score in FISH, it tends to produce colorful halos at the other edges. Thus, it gains below-average in PSNR and low in SSIM.

The T_TV method rank the fastest in processing time also it performed well in reduce image textures and its attempted to preserve important structure well. It keeps obvious contours in text regions. Therefore, it achieves above-average in PSNR and high scores in both SSIM and FISH. As for RGF method, it gave subpar performance in maintaining the vital edges and structure. It tends to blur the image structural details while removing its textures. That is why it gain low score in PSNR and below-average in both SSIM and FISH with slow processing time. The SSF method was better than RGF in maintaining the important edges, but in some cases, it didn't keep structural details well. that's why it gains a moderate FISH and SSIM scores and high score in PSNR. Also, it ranked the 5th in processing time. Although the RTV method ranked the second in processing time, its performance in preserving the image structure in smoothing process is not good enough also it blurs the edges. Therefore, it gains above-average in SSIM, average in PSNR, and low in FISH.

Although the proposed method ranked the 3rd in processing time, it fulfills perfect performance in term of PSNR, SSIM, and FISH. It achieved a good balancing between the important structural preserving and the texture funnel. It tends to maintain the fine internal structural details that lead to better semantic preservation and enhance the overall quality of the image. The proposed method was not shown edges breakage which leading to natural transactions in background without noticeable colorful halos, staircase, or other unwanted artifacts. The proposed algorithm has a limitation. It is not fully automated, adjusting the number of repetitions per image is subjective process that depends on the interpretation and perspective of the person performing the smoothing. Adjusting the number of repetitions per image is a challenge in itself. Furthermore, the hyper parameters () were tuned heuristically. Therefore, we will focus on future to make it fully automated.

4 STATISTICAL ANALYSIS

A statistical analysis is conducted in this section to assess and compare the performance of the proposed smoothing method with other methods through PSNR, SSIM, and FISH. The statistical summary in Table 5 describes the overall behaviour of each method using the mean, standard deviation, minimum and maximum values, and confidence intervals to show the stability and variability of the results. In addition, Table 6 presents the paired t-tests applied to compare the performance of the proposed method with other methods using the test statistic, p-value, confidence intervals, Average differences, and the magnitude of effect size.

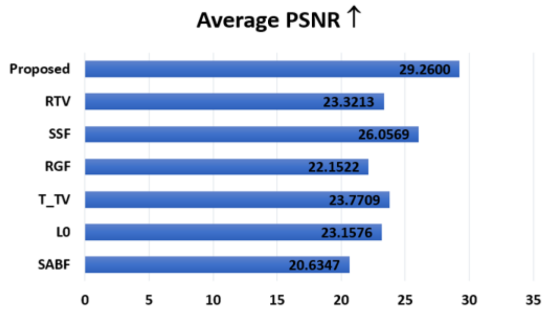

Figure 9: The recorded PSNR scores for comparison as chart.

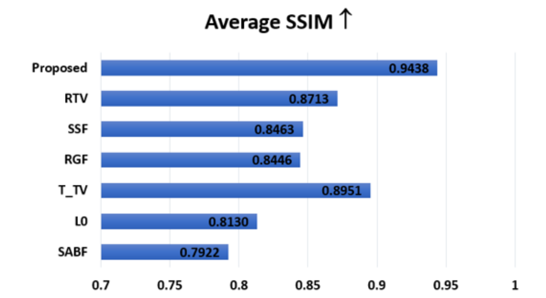

Figure 10: The recorded SSIM scores for comparison as chart.

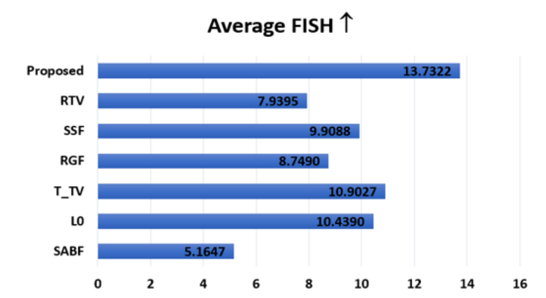

Figure 11: The recorded FISH scores for comparison as chart.

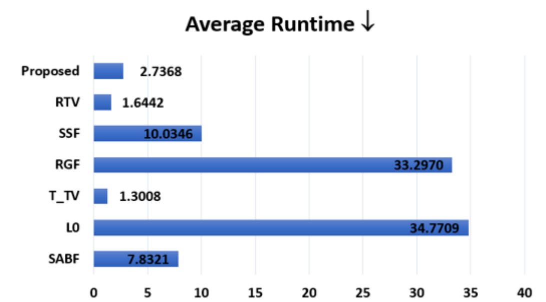

Figure 12: The recorded Runtime for comparison as chart.

The results in Table 5 show that the proposed smoothing method outperforms all the compared methods according to three evaluation metrics, FISH, PSNR, and SSIM. It achieves the highest PSNR value (≈30.96), indicating better image quality reconstruction compared to the best competing method (SSF ≈ 27.06). The proposed method also achieves higher SSIM values (≈ 0.90 compared with ≈ 0.69 for T_TV), showing that image structures are better preserved after smoothing.

In addition, the FISH results confirm the stronger preservation of important features and edges, with an average value of approximately 12.88, compared with 9.50 for the closest competitor. Overall, the small variability and the very significant statistical test results indicate that these improvements are stable and clearly support the advantage of the proposed method.

The paired t-test in Table 6 clearly shows that the proposed smoothing method performs better than all the compared methods for all metrics. For PSNR, the proposed method gives large improvements, ranging from about 3.9 units compared with SSF to more than 12 units compared with SABF, based on the average PSNR values. The paired t-test produces extremely small p-values indicating that these differences are not due to random variation. The very large effect size values also show that the improvement is strong and consistent across almost all images. Similar results are observed for SSIM, where the proposed method improves structural similarity and visual quality, with clear mean differences and highly significant test results. The FISH metric also confirms better preservation of image features and edges, with noticeable gains compared with all competing approaches. Overall, the paired t-test analysis strongly supports that the proposed method provides reliable, stable, and practically meaningful improvements in reconstruction quality, structural preservation, and feature retention across the dataset.

Overall, the results show that the proposed smoothing method performs better than all the compared methods for PSNR, SSIM, and FISH metrics. The statistical summary indicates that the proposed method gives higher image reconstruction quality, better preservation of structures and textures, and stronger protection of important features and edges across the 200 images. The paired t-test results also confirm that these improvements are highly significant, with very small adjusted p-values and large effect sizes, showing that the differences are not stochastic. In addition, the narrow confidence intervals and small variability indicate stable and reliable performance for different images. Therefore, the proposed method provides an effective smoothing solution by reducing noise while keeping important visual details.

Table 5: Statistical summary comparison of smoothing methods based on PSNR, SSIM, and FISH metrics over 200 images.

Metric	Method	Mean	Sd	Min	Max	Confidence Intervals	
						Lower bound	Upper bound
FISH	L0	8.795435	1.496295	5.3	12.026	8.586794	9.004076
	Proposed	12.88129	1.377357	9.931	15.481	12.68923	13.07335
	RGF	6.954905	1.55542	3.597	10.645	6.73802	7.17179
	RTV	5.987175	1.592881	2.425	9.712	5.765066	6.209284
	SABF	5.27268	1.631888	1.55	9.015	5.045132	5.500228
	SSF	7.9818	1.549319	4.617	11.634	7.765765	8.197835
	T_TV	9.49794	1.390378	6.318	12.378	9.304068	9.691812
PSNR	L0	21.87521	1.765203	17.001	26.097	21.62907	22.12135
	Proposed	30.96491	1.674349	27.438	34.527	30.73144	31.19837
	RGF	20.73422	1.761966	16.028	24.787	20.48853	20.97991
	RTV	22.73625	1.744357	18.049	26.84	22.49301	22.97948
	SABF	18.9304	1.872789	14.944	23.25	18.66926	19.19154
	SSF	27.06108	1.62274	23.602	30.691	26.83481	27.28735
	T_TV	23.62673	1.685507	19.648	27.674	23.3917	23.86175
SSIM	L0	0.605855	0.071383	0.385	0.743	0.595901	0.615809
	Proposed	0.902125	0.050312	0.731	0.984	0.89511	0.90914
	RGF	0.620235	0.07006	0.403	0.755	0.610466	0.630004
	RTV	0.664305	0.070717	0.449	0.808	0.654444	0.674166
	SABF	0.587965	0.070881	0.37	0.728	0.578081	0.597849
	SSF	0.64461	0.06961	0.43	0.782	0.634904	0.654316
	T_TV	0.69032	0.072143	0.461	0.827	0.68026	0.70038

Table 6: Paired t-Test statistical comparison between the proposed method and competing methods.

Metric	Compare	Statistic	P-value	Confidence intervals		Average difference	Cohen's effect size
				Lower bound	Upper bound		
PSNR	Proposed vs SABF	191.870	8.658E-228	11.910	12.158	12.034505	13.567
	Proposed vs L0	189.643	8.728E-227	8.995	9.184	9.089695	13.409
	Proposed vs T_TV	194.642	5.062E-229	7.263	7.412	7.33818	13.763
	Proposed vs RGF	207.615	1.428E-234	10.133	10.327	10.230685	14.680
	Proposed vs SSF	227.863	1.400E-242	3.870	3.937	3.903825	16.112
	Proposed vs RTV	179.967	2.759E-222	8.138	8.318	8.22866	12.725
SSIM	Proposed vs SABF	100.783	9.531E-173	0.308	0.320	0.31416	7.126
	Proposed vs L0	93.168	4.250E-166	0.289	0.302	0.29627	6.588
	Proposed vs T_TV	64.701	1.313E-135	0.205	0.218	0.211805	4.575
	Proposed vs RGF	91.838	6.974E-165	0.275	0.287	0.28189	6.493
	Proposed vs SSF	85.489	7.624E-159	0.251	0.263	0.257515	6.045
	Proposed vs RTV	76.568	1.338E-149	0.231	0.243	0.23782	5.414
FISH	Proposed vs SABF	87.907	3.417E-161	7.437	7.779	7.60861	6.215
	Proposed vs L0	60.899	1.266E-130	3.953	4.218	4.085855	4.306
	Proposed vs T_TV	82.412	9.195E-156	3.302	3.464	3.38335	5.827
	Proposed vs RGF	79.384	1.269E-152	5.779	6.073	5.926385	5.613
	Proposed vs SSF	66.978	1.821E-138	4.755	5.043	4.89949	4.736
	Proposed vs RTV	84.909	2.851E-158	6.734	7.054	6.894115	6.003

5 CONCLUSIONS

This study proposed a new smoothing method based on the Fractional Riesz Derivative (FRD) concept to smooth color image and achieving a balance between the image smoothing and preserves its essential structural. It enhanced the image through iterative process via the FRD for image luminance, adaptive smoothing weights, and multi-channel smoothing. The FRD is used to extract the high frequency component, then an adaptive smoothing weights is used to avoid the excessive smoothing at edges and make the transition between the edge and the flat area became more gradually, finally, a multi-channel smoothing which include applying a normalized smoothing on each color channel separately to prevent color shift and to ensure the preserving of image mean brightness. The proposed method was tested using various nature images from COCO data set, compared with six other methods and evaluated using three measures in addition to the processing time. The presented outcomes demonstrate the capability of the proposed method in suppression the texture and preserve the essential structural details without causing any unwanted artifacts. In the forthcoming, we will focus on providing an adaptive method to makes the proposed algorithm fully automated, through determining the appropriate number of repetitions per each image that achieve superior results.

ACKNOWLEDGMENTS

The authors are grateful to the College of Education for Pure Science at the University of Mosul in Iraq for their support.

REFERENCES

[1] W. Zhao, T. Wu, CH. Feng, W. Wu, X. Lv, H. Chen, and J. Liu, "Image Smoothing via a Novel Adaptive Weighted L0 Regularization," Int. J. Numer. Anal. Model., vol. 22, no. 1, pp. 21-39, Nov. 2024, doi: 10.4208/ijnam2025-1002.

[2] L. Jiang, Y. Han, and B. Xie, "Edge-Preserving Image Smoothing Via a Total Variation Regularizer and a Nonconvex Regularizer," Procedia Comput. Sci., vol. 154, pp. 603-609, 2019, doi: 10.1016/j.procs.2019.06.095.

[3] J. Long, Q. Luo, S. Chen, and Y. Liu, "Edge-preserving image smoothing algorithm combining edge guidance and texture complexity perception," Digit. Signal Process., vol. 169, p. 105710, Feb. 2026, doi: 10.1016/j.dsp.2025.105710.

[4] M. A. Al-Abaji and Z. Al-Ameen, "Contrast enhancement of MR images using S-curve transforms with logarithmic image processing," 2025, p. 040018, doi: 10.1063/5.0259226.

[5] C. Tomasi and R. Manduchi, "Bilateral Filtering for Gray and Color Images," in Sixth International Conference on Computer Vision, IEEE, 1998, pp. 839-846.

[6] G. Petschnigg, M. Agrawala, H. Hoppe, R. Szeliski, M. Cohen, and K. Toyama, “Digital photography with flash and no-flash image pairs,” ACM Trans. Graph., vol. 23, no. 3, pp. 664-672, 2004.

[7] K. He, J. Sun, and X. Tang, “Guided Image Filtering,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 35, no. 6, pp. 1397-1409, 2012, doi: 10.1109/TPAMI.2012.213.

[8] Z. Li, J. Zheng, Z. Zhu, W. Yao, and S. Wu, “Weighted Guided Image Filtering,” IEEE Trans. Image Process., vol. 24, no. 1, pp. 120-129, 2015, doi: 10.1109/TIP.2014.2371234.

[9] Q. Zhang, X. Shen, L. Xu, and Jiaya Jia, “Rolling Guidance Filter,” in European Conference on Computer Vision, Springer, Cham, 2014, pp. 815-830, [Online]. Available: https://doi.org/10.1007/978-3-319-10578-9_53.

[10] H. Qi, X. Luo, T. Li, and F. Li, “EDGE-AWARE IMAGE SMOOTHING WITH RELATIVE WAVELET DOMAIN REPRESENTATION,” in The Thirteenth International Conference on Learning Representations (ICLR 2025), 2025.

[11] L. Xu, C. Lu, Y. Xu, and J. Jia, “Image Smoothing via L0 Gradient Minimization,” in Proceedings of the 2011 SIGGRAPH Asia Conference (December 2011), 2011, pp. 1-12, [Online]. Available: https://doi.org/10.1145/2024156.2024208.

[12] L. Xu, Q. Yan, Y. Xia, and J. Jia, “Structure Extraction from Texture via Relative Total Variation,” ACM Trans. Graph., vol. 31, no. 6, pp. 1-10, 2012.

[13] B. Ham, M. Cho, and J. Ponce, “Robust Guided Image Filtering Using Nonconvex Potentials,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 40, no. 1, pp. 192-207, 2017.

[14] W. LIU, P. ZHANG, X. HUANG, J. YANG, C. SHEN, and I. REID, “Real-time Image Smoothing via Iterative Least Squares,” ACM Trans. Graph., vol. 39, no. 3, pp. 1-24, 2020.

[15] W. Liu, P. Zhang, Y. Lei, X. Huang, J. Yang, and M. Ng, “A Generalized Framework for Edge-Preserving and Structure-Preserving Image Smoothing,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 44, no. 10, pp. 6631-6648, 2021, [Online]. Available: https://doi.org/10.48550/arXiv.2107.07058.

[16] J. Huang, H. Wang, X. Wang, and M. Ruzhansky, “Semi-Sparsity for Smoothing Filters,” IEEE Trans. Image Process., vol. 32, pp. 1627-1639, 2023, doi: 10.1109/TIP.2023.3247181.

[17] Y. Yang, D. Wu, L. Tang, L. Zeng, and Z. Pan, “Weighted and truncated L1 image smoothing based on unsupervised learning,” Vis. Comput., vol. 40, pp. 5871-5882, 2024, doi: 10.1007/s00371-023-03141-0.

[18] H. Qi, S. Tan, and X. Luo, “Self-Supervised Dual Generative Networks for Edge-Preserving Image Smoothing,” in 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2024, doi: 10.1109/ICASSP48485.2024.10448288.

[19] J. Pan, D. Sun, J. Zhang, J. Tang, J. Yang, Y. Tai, and M. Yang, “Dual Convolutional Neural Networks for Low-Level Vision,” Int. J. Comput. Vis., vol. 130, pp. 1440-1458, 2022, [Online]. Available: https://doi.org/10.1007/s11263-022-01583-y.

[20] M. Unser and T. Blu, “Fractional Splines and Wavelets,” SIAM Rev., vol. 42, no. 1, pp. 43-67, Jan. 2000, doi: 10.1137/S0036144598349435.

[21] W. T. Freeman and E. H. Adelson, “The design and use of steerable filters,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 13, no. 9, pp. 891-906, 1991, doi: 10.1109/34.93808.

[22] P. Perona and J. Malik, “Scale-space and edge detection using anisotropic diffusion,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 12, no. 7, pp. 629-639, Jul. 1990, doi: 10.1109/34.56205.

[23] T.-Y. Lin, M. Maire, S. Belongie, L. Bourdev, R. Girshick, J. Hays, P. Perona, D. Ramanan, C. L. Zitnick, and P. Doll´ar, “Microsoft COCO: Common Objects in Context,” in Lecture Notes in Computer Science, Springer, 2014, [Online]. Available: https://doi.org/10.1007/978-3-319-10602-1_48.

[24] S. Ghosh, R. G. Gavaskar, D. Panda, and K. N. Chaudhury, “Fast Scale-Adaptive Bilateral Texture Smoothing,” IEEE Trans. Circuits Syst. Video Technol., pp. 1-1, 2019, doi: 10.1109/TCSVT.2019.2916589.

[25] Z. Dou, M. Song, K. Gao, and Z. Jiang, “Image Smoothing via Truncated Total Variation,” IEEE Access, vol. 5, pp. 27337-27344, 2017, doi: 10.1109/ACCESS.2017.2773503.

[26] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, “Image quality assessment: from error visibility to structural similarity,” IEEE Trans. Image Process., vol. 13, no. 4, pp. 600-612, 2004, doi: 10.1109/TIP.2003.819861.

[27] P. V. Vu and D. M. Chandler, “A Fast Wavelet-Based Algorithm for Global and Local Image Sharpness Estimation,” IEEE Signal Process. Lett., vol. 19, no. 7, pp. 423-426, 2012, doi: 10.1109/LSP.2012.2199980.