Journal of Imaging Science and Technology® 67(6): 1–18, 2023.
© Society for Imaging Science and Technology 2023

Spatio–Temporal Retinex–Inspired Envelopes with Anisotropic Diffusion

Petter Sagvold, Ivar Farup[▲], and Marius Pedersen[▲]
Colourlab Department of Computer Science, Norwegian University of Science and Technology (NTNU), Gjøvik, Norway
E-mail: ivar.farup@ntnu.no

Abstract. *Since the introduction of the Retinex theory by Land and McCann in 1971, a multitude of different families, versions, interpretations, implementations, and applications have been proposed. The applications for image enhancement mainly differ in (i) how they explore the locality of the images to determine the local context, and (ii) how they recompute the pixel values based on this context. STRESS (spatio-temporal Retinex-inspired envelopes with stochastic sampling) is one of many quite successful members of the family of Retinex-based image enhancement algorithms. It explores the locality using a stochastic sampling technique, resulting in two envelope images – one maximum and one minimum envelope, completely enclosing the image signal and serving as a representation of the local image context. In this paper, we propose to exchange the stochastic sampling technique of STRESS, which causes significant chromatic noise, with an adapted version of constrained linear anisotropic diffusion for computing the envelopes, resulting in almost noise-free images. Using both subjective experiments and objective image metrics, we show that it improves the perceived and measured image quality and reduces noise artefacts.* © *2023 Society for Imaging Science and Technology.*
[DOI: 10.2352/J.ImagingSci.Technol.2023.67.6.060407]

1. INTRODUCTION

The primary purpose of image enhancement is to improve the perceived quality of an image for a human observer. Many techniques have been proposed over the years, with varying effectiveness and efficiency. A common strategy for many algorithms is that they in one way or another, try to mimic some properties of the human visual system. Many such algorithms are based on the Retinex theory of colour vision [1–3]. Retinex-based image enhancement is often a two-step procedure. First, the local context of the image is computed. Second, the pixel values are recomputed based on the local context. Many families, versions, interpretations, and implementations of Retinex-based image enhancement techniques have been proposed over the years, and it has been used for various colour imaging applications such as colour correction, computational colour constancy, HDR image rendering [4], colour gamut mapping, and colour-to-greyscale conversion [5]. More recent developments take Retinex-based methods in the direction of deep learning [6–8].

STRESS – Spatio-temporal Retinex-inspired envelopes with stochastic sampling – is one quite successful member of the family of Retinex-inspired algorithms [9]. In STRESS, the locality is represented by two envelope colour images with gamma-corrected RGB values, $E_{\max}$ and $E_{\min}$, which have the properties $0 \leq E_{\min} \leq u_0 \leq E_{\max} \leq u_0$, where u_0 is the original image. Similar to other Retinex-based algorithms such as RSR [10], these envelopes are computed by a stochastic sampling technique, and are thus subject to chromatic noise. The noise in each channel is caused by random sampling, and its chromatic content is due to the independent sampling of the three colour channels. In the second step, the (gamma-corrected) values of the original image are linearly rescaled using the envelopes.

In this paper, we propose another method for exploring the locality in the STRESS algorithm to significantly reduce the noise caused by the stochastic sampling. Similar to more recent methods like STRETV (based on total variation, which is isotropic) [11] and ReMark (based on Markov chains, also isotropic) [12], we focus our attention on diffusion-based approach. By introducing an adapted and constrained version of linear *anisotropic* diffusion – a technique originally aimed at reducing image noise that has recently been applied to colour imaging applications beyond denoising [13, 14] – for computing the envelopes, we can minimise the image noise resulting from the noise in the envelopes.

In Section 2, we present the basic ideas of the STRESS algorithm and anisotropic diffusion. Then, in Section 3, we detail the proposal of using anisotropic diffusion for computing the envelopes and show example results. The experimental setup for evaluating its performance both in terms of overall image preference and noise is described in Section 4, and the results are given in Section 5, before concluding in Section 6.

2. BACKGROUND

2.1 *STRESS*

Some Retinex-based implementations explore the images by using paths or computing ratios with neighbours in a multilevel framework [3, 15–19] or using models of Brownian motions to analyse the image along paths [20, 21]. Other implementations compute values over the given image with convolution masks or weighting distances [22–26]. In a

[▲] IS&T Members.

Received July 14, 2023; accepted for publication Dec. 7, 2023; published online Jan. 3, 2024. Associate Editor: Francisco Imai.
1062-3701/2023/67(6)/060407/18/$25.00

study [10], the path-based scanning was substituted by a new approach using random sampling of a cloud of points.

A similar approach was followed for the STRESS algorithm [9]. Here, the visual context was characterised using two envelopes $E_{\max}$ and $E_{\min}$. For each pixel, p_0, the values of the maximum and minimum envelopes at the corresponding positions, are computed through N iterations. In every iteration, M pixel values p_i in each channel are sampled at random with a probability proportional to $1/d$, d being the Euclidean distance in the image from the sampled pixel to the pixel in question. The value of the centre pixel is not eligible for random sampling but is always included in the sampled set. From these samples, the maximum and minimum samples in the spray are found: $s_{\max} = \max(p_j)$, $s_{\min} = \min(p_j)$. Since p_0 is always among the sample points, it is granted that $s_{\min} \leq p_0 \leq s_{\max}$. These maximum and minimum points could be taken as direct estimates for the envelopes. However, better results were achieved when the relative position v_i of the pixel p_0 within the range $r_i = s_{\max} - s_{\min}$ was used.

The final envelopes were computed as

$$E_{\min} = p_0 + \bar{v}\bar{r} \tag{1}$$

$$E_{\max} = E_{\min} + \bar{r}, \tag{2}$$

where $\bar{v}$ is the average of the v values, and $\bar{r}$ is the average of the r values over the N iterations. The new image is recomputed by stretching the image to these envelopes,

$$p = \frac{p_0 - E_{\min}}{E_{\max} - E_{\min}} \tag{3}$$

It should be noted that this is a heavy computational technique requiring $\mathcal{O}(NMP)$ operations, where N is the number of iterations, M is the number of samples per iteration, and P is the number of pixels in the image.

The sampling technique also introduces a significant chromatic noise. The type of noise is quite particular to STRESS. To the best of our knowledge, this type of noise has not been characterised in the literature. A histogram of the noise in each colour channel, produced by computing the difference between STRESS images with 100 (noisy) versus 1000 (virtually noiseless) iterations, is shown in Figure 1. It is symmetric, but far from Gaussian. There is no correlation between the noise of the different image channels.

2.2 *Anisotropic Diffusion*

Since this method produces significant chromatic noise, another approach is taken in the STRETV algorithm [11]. Here, the constrained total variation method is used to calculate the envelopes. Total variation minimisation results in a process very similar to non-linear isotropic diffusion. This implementation showed promising results when used in contrast enhancement and in automatic colour correction, with significantly reduced noise levels. However, due to the non-linearity of total variation, the method requires a tiny time-step and thus many iterations to converge. Although it behaves nicely near edges in the original image, it creates some artefacts near corners and lines of high curvature.

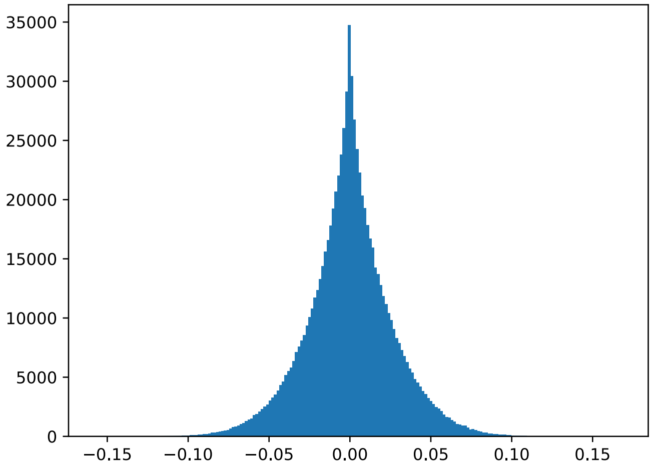


Figure 1. Histogram of the noise caused by random sampling in the STRESS algorithm.

Various diffusion techniques have been widely used in computer vision and image processing to reduce image noise without removing significant information from the image. One important technique is the Perona–Malik diffusion [27], which is an isotropic, local, non-linear diffusion technique, not too different from total variation minimisation. Unfortunately, Perona–Malik diffusion was misnomed “anisotropic diffusion”. Real anisotropic diffusion for image processing was first described by Tschumperle and Deriche [28] as a non-linear process.

The starting point for Tschumperle and Deriche non-linear anisotropic diffusion is the 2 × 2 structure tensor S of the original image [29], whose components for every single pixel can be expressed as

$$S_{ij} = \sum_{\mu} \frac{\partial u_0^{\mu}}{\partial x^i} \frac{\partial u_0^{\mu}}{\partial x^j}. \tag{4}$$

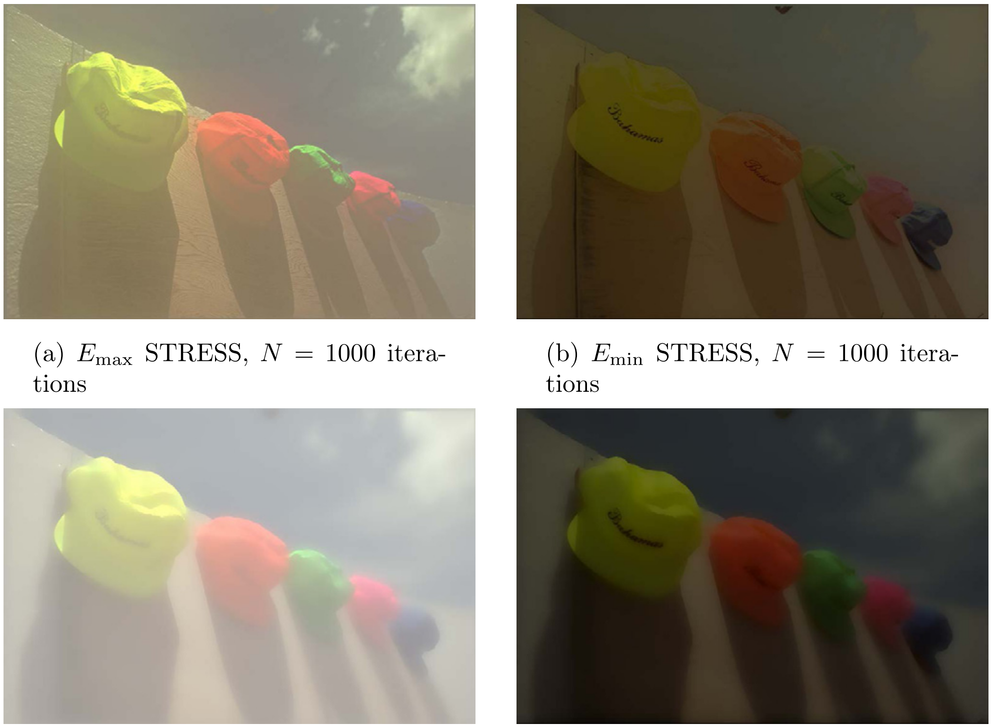
Here u_0 denotes the original image, the indexes μ and ν denote the colour channels, and x^i and x^j denote the two spatial directions. The eigenvalues of the structure tensor S are denoted λ^+ and λ^-, and the corresponding normalised eigenvectors $\mathbf{e}^+$ and $\mathbf{e}^-$ are stored as columns in the orthonormal eigenvector matrix V, such that the structure tensor can be written $S = V^T \mathrm{diag}(\lambda^+, \lambda^-) V$ [14]. From this, the diffusion tensor is then defined as

$$D = V^T \mathrm{diag}(d(\lambda^+), d(\lambda^-)) V, \tag{5}$$

where $d(\lambda)$ is a nonlinear diffusion coefficient function (Eq. (6)) [14] whose task is to suppress the diffusion across the edges while preserving it along the edges,

$$d(\lambda) = \frac{1}{1 + \kappa \lambda^2} \tag{6}$$

and κ is a suitably chosen numeric constant. Higher values of κ will give more edge preservation in the image.



(a) $E_{\max}$ STRESS, $N = 1000$ iterations

(b) $E_{\min}$ STRESS, $N = 1000$ iterations

(c) $E_{\max}$ STREAD, $N = 1000$ iterations

(d) $E_{\min}$ STREAD, $N = 1000$ iterations

Figure 2. Envelopes: $\kappa = 1000$ for STREAD, $M = 3$ sampling points for STRESS.

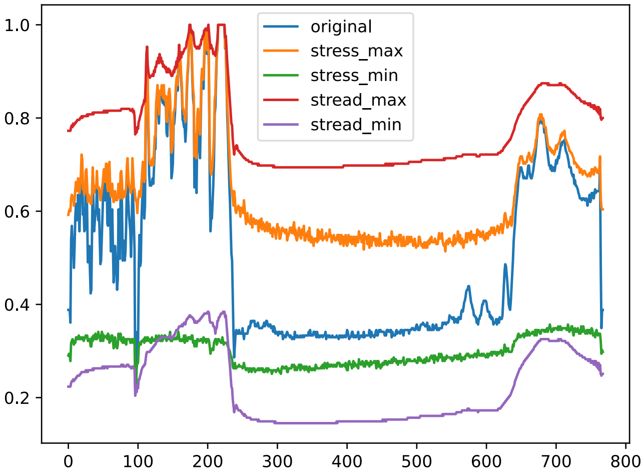


Figure 3. Scan lines for the envelopes shown in Figure 2.

Having defined the diffusion tensor, D, anisotropic diffusion results from solving the Euler–Lagrange equations for minimising its eigenvalues by gradient descent using the artificial time parameter t (corresponding to the iterations when discretised),

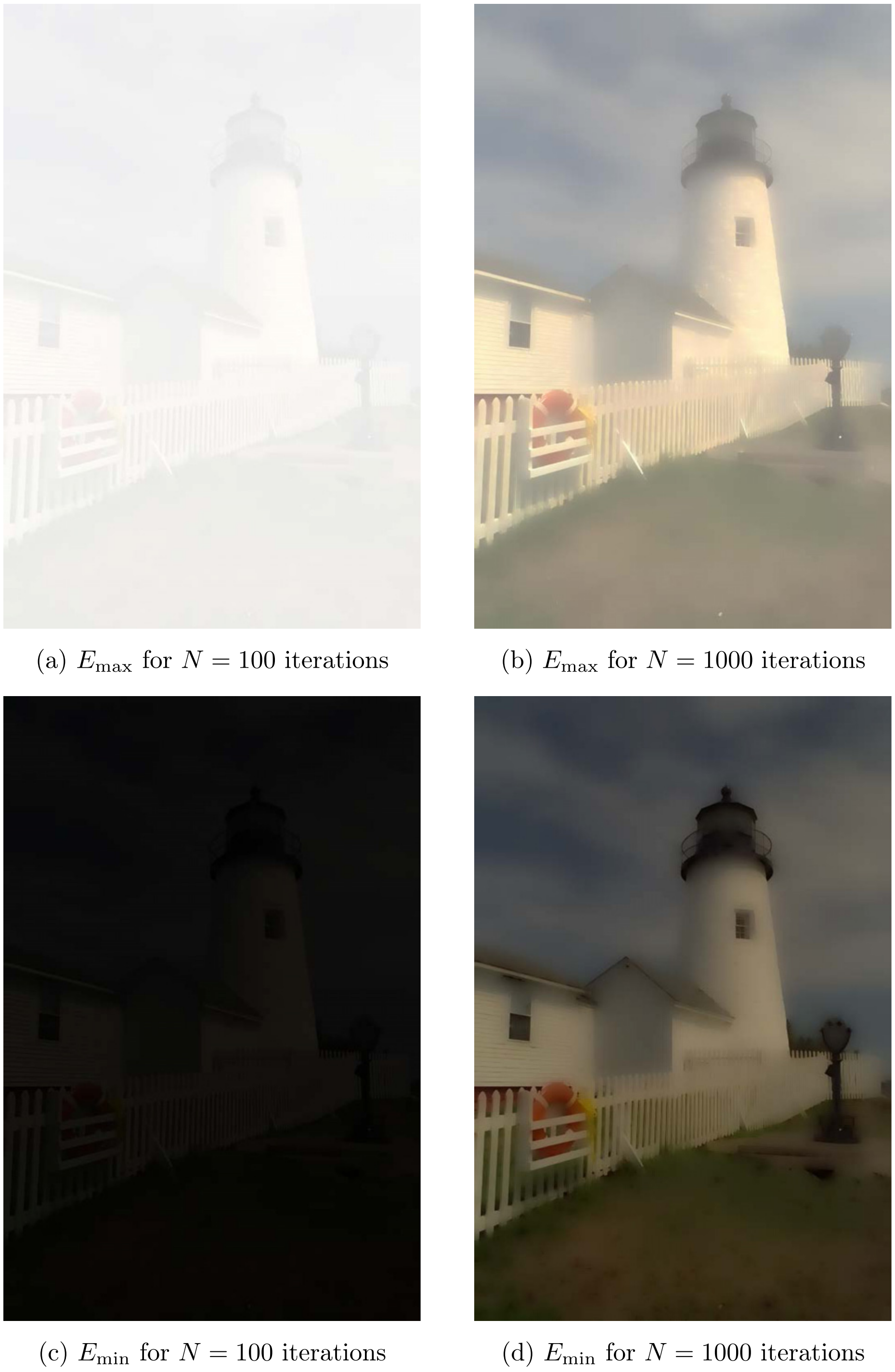
$$\frac{\partial u}{\partial t} = \nabla \cdot (D \nabla u). \tag{7}$$

It has been ascertained in various studies (see, e.g., [13]) that the structure tensor, and thus the diffusion tensor, can be computed once and for all from the original image. Then, the diffusion equation, Eq. (7) becomes linear. In the same study, it was also found that the anisotropic diffusion was better than the isotropic one (Perona–Malik-type) for preserving edges and, in particular, corners.

It should be noted that, even after linearisation, the solution of the anisotropic diffusion equation employing iterative gradient descent is, like STRESS, a computationally heavy procedure. The diffusion lengths, and thus the number of iterations needed, are based on the image size, and the diffusion tensor that reduces the diffusion locally increases the need for iterations further.

3. PROPOSED ALGORITHM

We introduce a new model for spatio-temporal image enhancement called STREAD (Spatio-Temporal Retinex-Inspired Envelope with Anisotropic Diffusion). This model is based on the STRESS algorithm, which has a main feature of computing the envelopes $E_{\max}$ and $E_{\min}$ for each channel of the image. However, instead of applying stochastic sampling to obtain $E_{\max}$ and $E_{\min}$ as in STRESS, we use anisotropic diffusion [28]. The second part of the algorithm, i.e., the recomputation of the pixel values based on the computed envelopes, remains unaltered.



(a) $E_{\max}$ for $N = 100$ iterations

(b) $E_{\max}$ for $N = 1000$ iterations

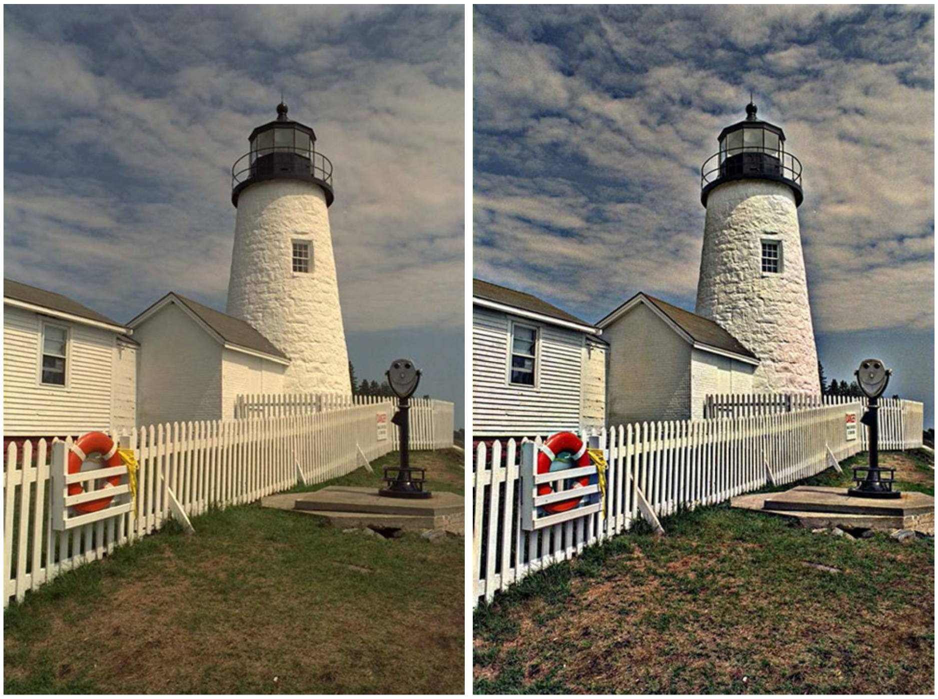
(c) $E_{\min}$ for $N = 100$ iterations

(d) $E_{\min}$ for $N = 1000$ iterations

Figure 4. STREAD envelopes for $N \in \{100, 1000\}$ iterations, $\kappa = 1000$.

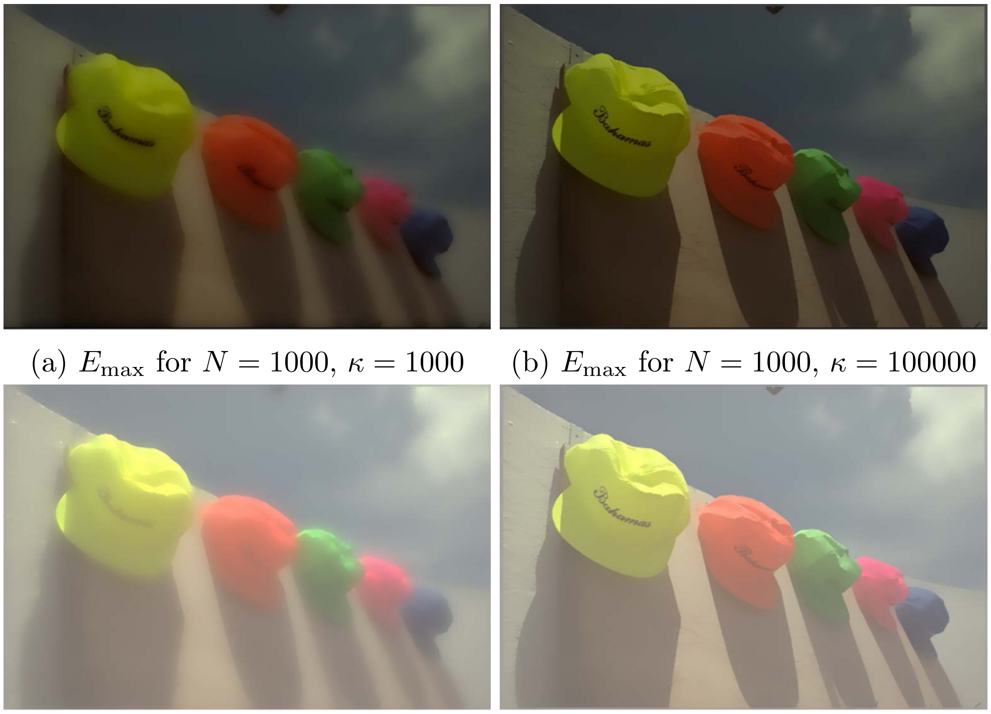
3.1 *Anisotropic Diffusion for Computing the Envelopes*

The envelopes of STRESS are images $E_{\max}$ and $E_{\min}$ with the property that, for each image channel in the original image u_0, $0 \leq E_{\min} \leq u_0 \leq E_{\max} \leq 1$. The basic idea here, is to use the diffusion tensor of anisotropic diffusion, Eq. (5) to compute alternative versions of the STRESS envelopes. This can be achieved in two different ways. Either, the original image is used as the initial value for both envelopes, or a black image is used as the initial value for the minimum envelope and a white image for the maximum one. In both cases, adding a data attachment term to the diffusion equation will ensure that the envelopes will stay reasonably close to the



(a) $N = 100$ iterations (b) $N = 1000$ iterations

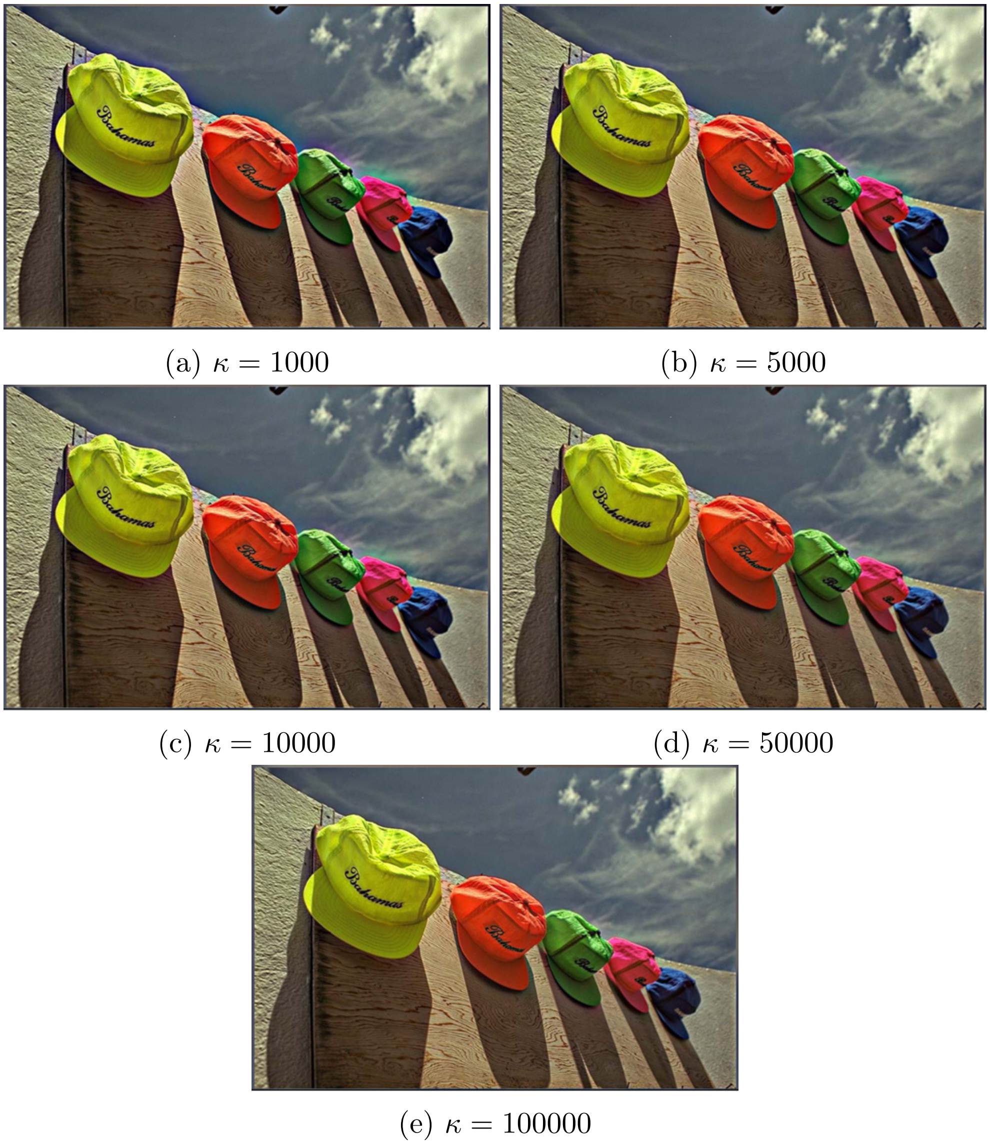
Figure 5. STREAD: 100/1000 iterations, $\kappa = 1000$.



(a) $E_{\max}$ for $N = 1000$, $\kappa = 1000$ (b) $E_{\max}$ for $N = 1000$, $\kappa = 100000$

(c) $E_{\min}$ for $N = 1000$, $\kappa = 1000$ (d) $E_{\min}$ for $N = 1000$, $\kappa = 100000$

Figure 6. Envelopes for $\kappa = 1000$ and $\kappa = 100000$.


(a) $\kappa = 1000$

(b) $\kappa = 5000$

(c) $\kappa = 10000$

(d) $\kappa = 50000$

(e) $\kappa = 100000$

Figure 7. Images with different values of κ.

image. The resulting equations for the envelopes are

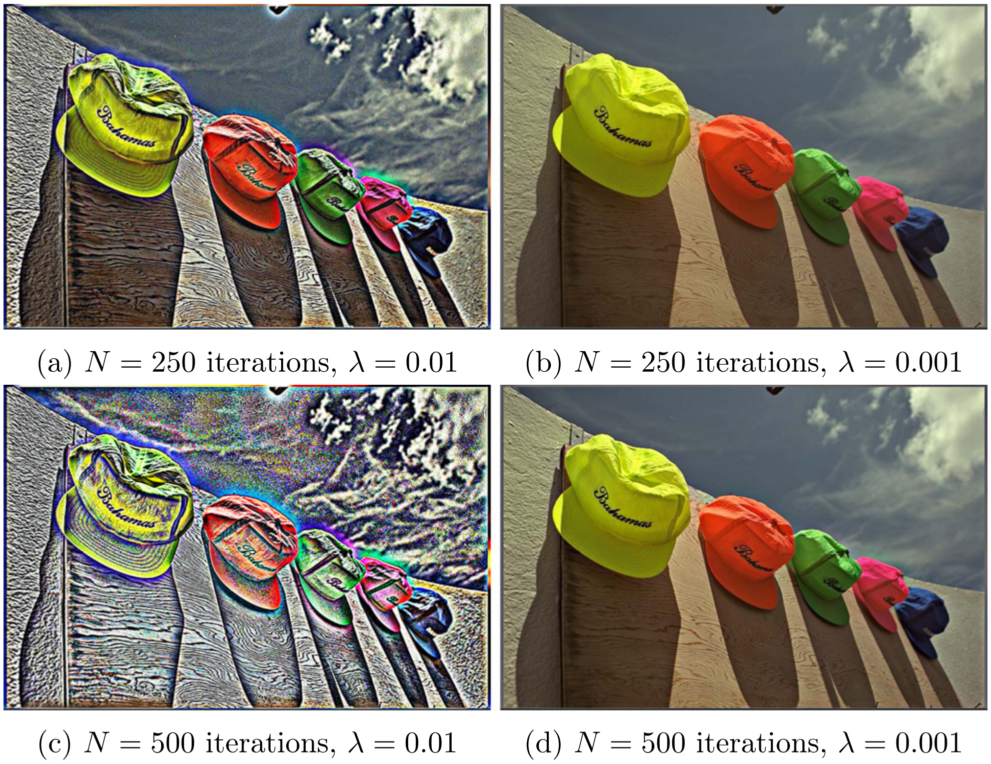
$$\frac{\partial E_{\max}}{\partial t} = \nabla(D\nabla E_{\max}) - \lambda(E_{\max} - u_0) \quad \text{s.t.} \quad E_{\max} \geq u_0 \tag{8}$$

$$\frac{\partial E_{\min}}{\partial t} = \nabla(D\nabla E_{\min}) - \lambda(E_{\min} - u_0) \quad \text{s.t.} \quad E_{\min} \leq u_0. \tag{9}$$

Normal boundary conditions are applied to the envelopes to make sure that there are no problems when calculating with the pixels at the border of the image. The equations are solved by the explicit Euler method for the time integration, and centred differences for the spatial derivatives. For each iteration, $E_{\max}$ and $E_{\min}$ are constrained so that they are greater than or smaller than the original image, respectively. The envelopes and the original image are finally used to recompute the image exactly as in the original STRESS algorithm.

3.2 *Resulting Envelopes*

In Fig. 2, the envelopes for STRESS and STREAD are shown. The envelopes contain some of the original image content, always brighter (for $E_{\max}$) or darker (for $E_{\min}$), and preserve the edges of the image. A plot of the corresponding scan lines is shown in Figure 3. One can see that the envelopes of STREAD are smoother than those of STRESS, and also that they, in general, are not so close to the original image. This means that STREAD will lead to less dramatic changes of the images. Moreover, both algorithms can follow sharp edges in the image (left part of the graphs). The less sharp transitions for the STREAD envelpes on the right part of the graph can be explained by other image information transversal to the scan line that diffuses into the shown line of the image.


(a) $N = 250$ iterations, $\lambda = 0.01$

(b) $N = 250$ iterations, $\lambda = 0.001$

(c) $N = 500$ iterations, $\lambda = 0.01$

(d) $N = 500$ iterations, $\lambda = 0.001$

Figure 8. Images with different values of λ and N. $\kappa = 1000$, $\lambda \in \{0.01, 0.001\}$, and $N \in \{250, 500\}$.

3.3 *Impact of Parameters*

As with the STRESS algorithm, the behaviour of STREAD varies with the values of its parameters. The influence of these parameters on the output of the algorithm is explained in detail.

Although both STRESS and STREAD converge to the final solution, iterations in STRESS and STREAD have quite different effects. STRESS iterates the stochastic sampling and computes the average to reduce the sampling chromatic noise, while iterations in STREAD leads to the stationary solution of the constrained PDEs.

The algorithm was tested with $N \in \{100, 200, 300, 400, 500, 600, 700, 800, 900, 1000\}$ iterations. With this test, it was easier to see the influence of higher iterations on the output. In Figure 4 the envelopes are shown for the extreme values of N in this set, and Figure 5 shows the resulting images. The edges are much sharper in the image with the most iterations. One can also notice some artefacts starting to appear near the edges.

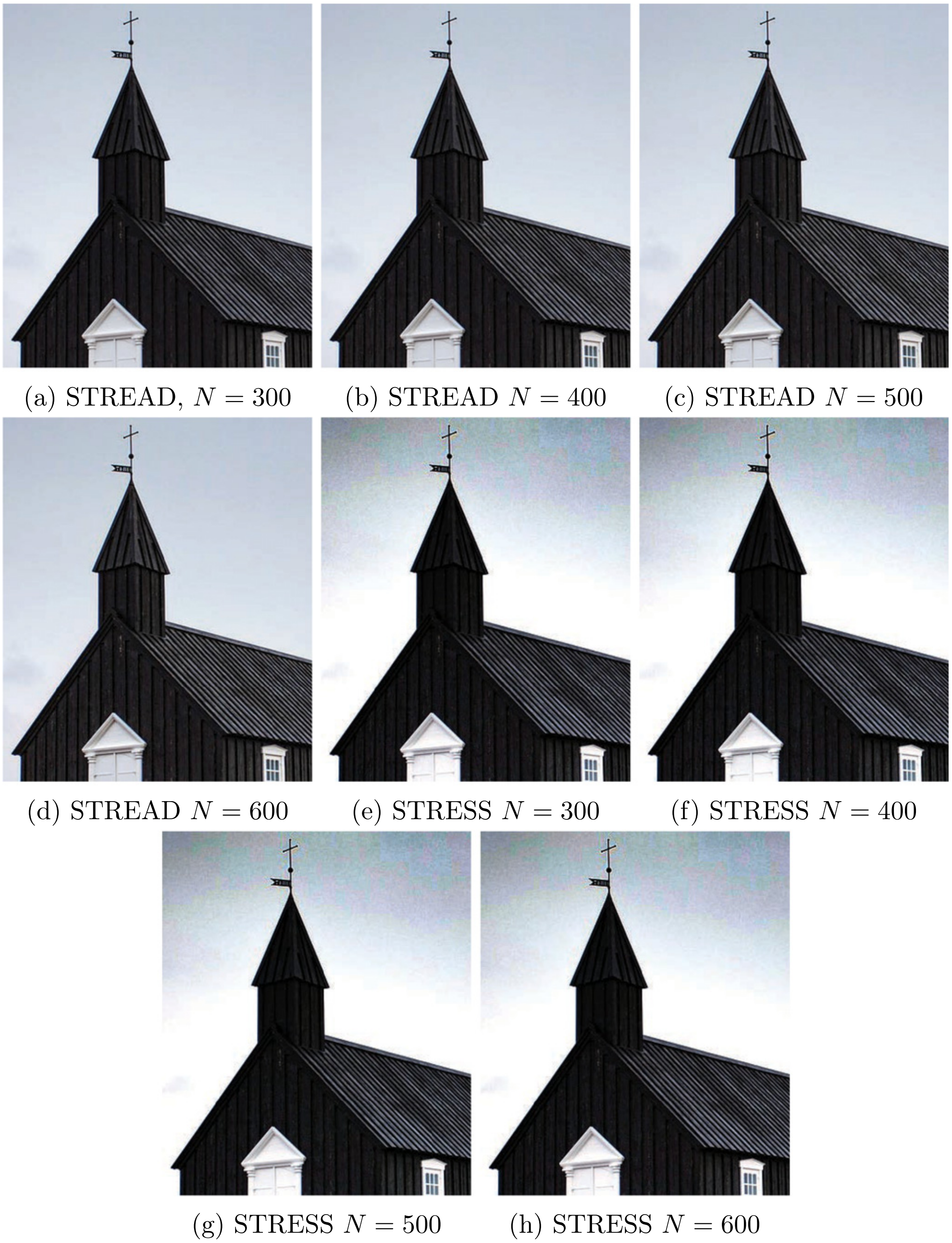
As for the iterations, experiments were performed with κ to determine what value it had to be to give the best output. While testing the highest value for iterations, a value of $\kappa = 1000$ was used. When the number of iterations became higher than 700, artefacts like halos started to appear on edges in the image. Results for different values of κ are shown in Figures 6 and 7 for the envelopes and resulting images, respectively. In Fig. 7(a), we can see how the halos appear around the edges in the image (see, e.g., the greenish area in the sky close to the red/pink caps, number four from the left). To counter this, a higher value of κ is needed, since higher values of κ will give more edge preservation in the image. Even with $\kappa = 100000$ and $N = 1000$, some artefacts still appear around some edges, but lower number of iterations does not result in these artefacts. Based on this, κ was set to 10000 and $N \in \{300, 350, 400, 450, 500, 550, 600\}$ for the remaing studies.

The last variable is λ, which is used to control the strength of the data attachment. The data attachment term $\lambda(u - u_0)$ in Eqs. (8) and (9) is a regularisation term that incorporates the prior information about the original image into the enhancement process, and acts as a constraint that minimises the discrepancy between the envelope and the original image [30, 31]. If $\lambda = 0$, the envelopes will be flat, resulting in the algorithm becoming spatially independent or global. If $\lambda \to \infty$, the envelopes will be the same as the original image. Results for different values of λ can be seen in Figure 8.

Figure 9 shows a comparison between STREAD and STRESS for different numbers of iterations. It can be seen that STREAD is better at enhancing the vertical lines on the wall. STRESS also amplified the noise present in the more uniform areas.

3.4 *Colour Balance*

One challenge with the preliminary experiments was that while STREAD kept more or less the same colour balance as the original, STRESS made the images a bit brighter and



(a) STREAD, $N = 300$ (b) STREAD $N = 400$ (c) STREAD $N = 500$

(d) STREAD $N = 600$ (e) STRESS $N = 300$ (f) STRESS $N = 400$

(g) STRESS $N = 500$ (h) STRESS $N = 600$

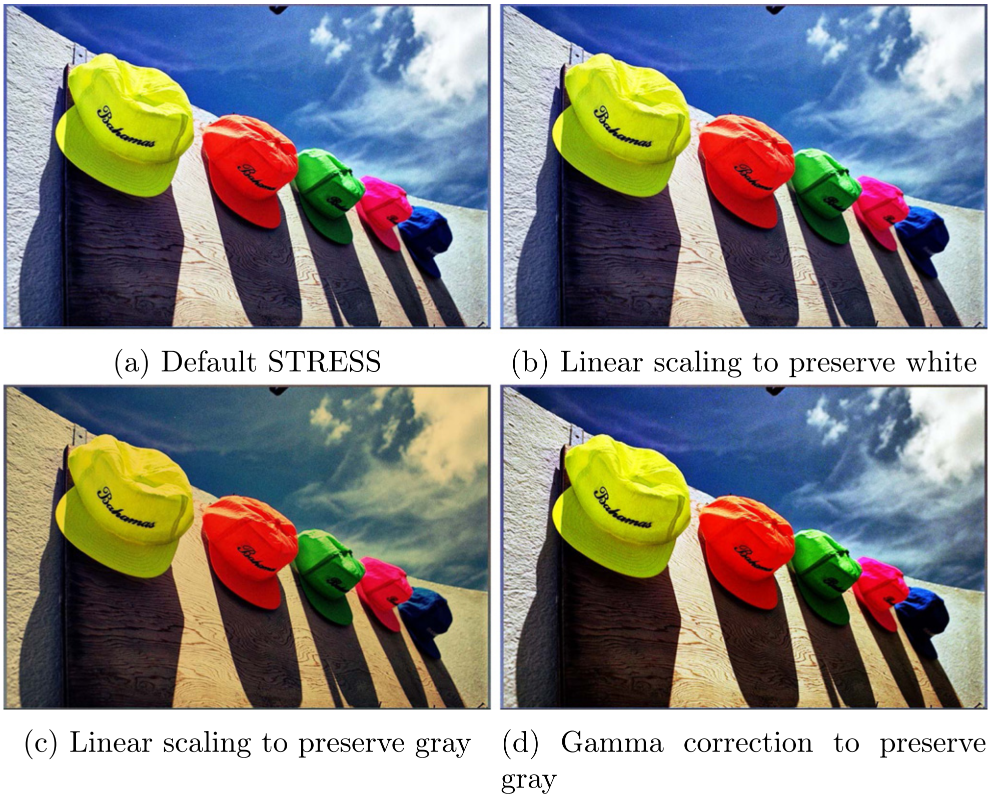
Figure 9. Comparison between STREAD and STRESS for different iterations. The images have been cropped to better show differences.

a bit blue, see Figure 11. In order for the images to become more comparable, the last stage of STRESS was modified to counteract this. Two linear scalings for preserving white and gray and a gamma correction for preserving gray were added. The difference between these two can be seen in Figure 10. Linear scaling to preserve gray was chosen, as it was the one that looked most similar to the original and STREAD. The scaling is performed for each channel separately by multiplying the result with the mean of the original image, and then dividing it by the mean of the resulting image. After these preliminary experiments, the final image set was chosen.

4. SUBJECTIVE EXPERIMENT

4.1 *Comparing STREAD with STRESS*

In a preliminary experiment, we compared visually the images resulting from STREAD to those of STRETV [11], and found that the STREAD images were so much better than the STRETV ones, that STRETV was left out of the experiments. The images generated by STREAD and STRESS were compared using a subjective survey and objective image metrics. Initial tests indicated that STREAD performed well with $\kappa = 10000$, $\lambda = 0.001$ and N ranging from 300 to 600. A sample of 10 standard RGB test images was selected for a paired comparison evaluation. The selected images are



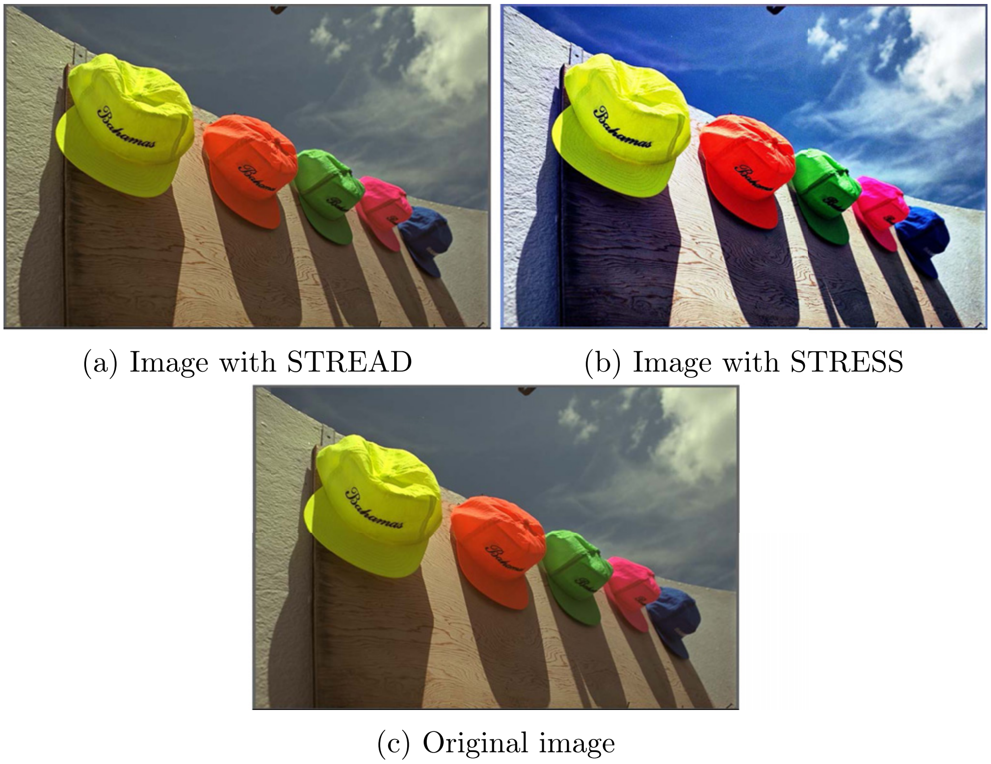
(a) Default STRESS

(b) Linear scaling to preserve white

(c) Linear scaling to preserve gray

(d) Gamma correction to preserve gray

Figure 10. Colour-corrected images for STRESS.

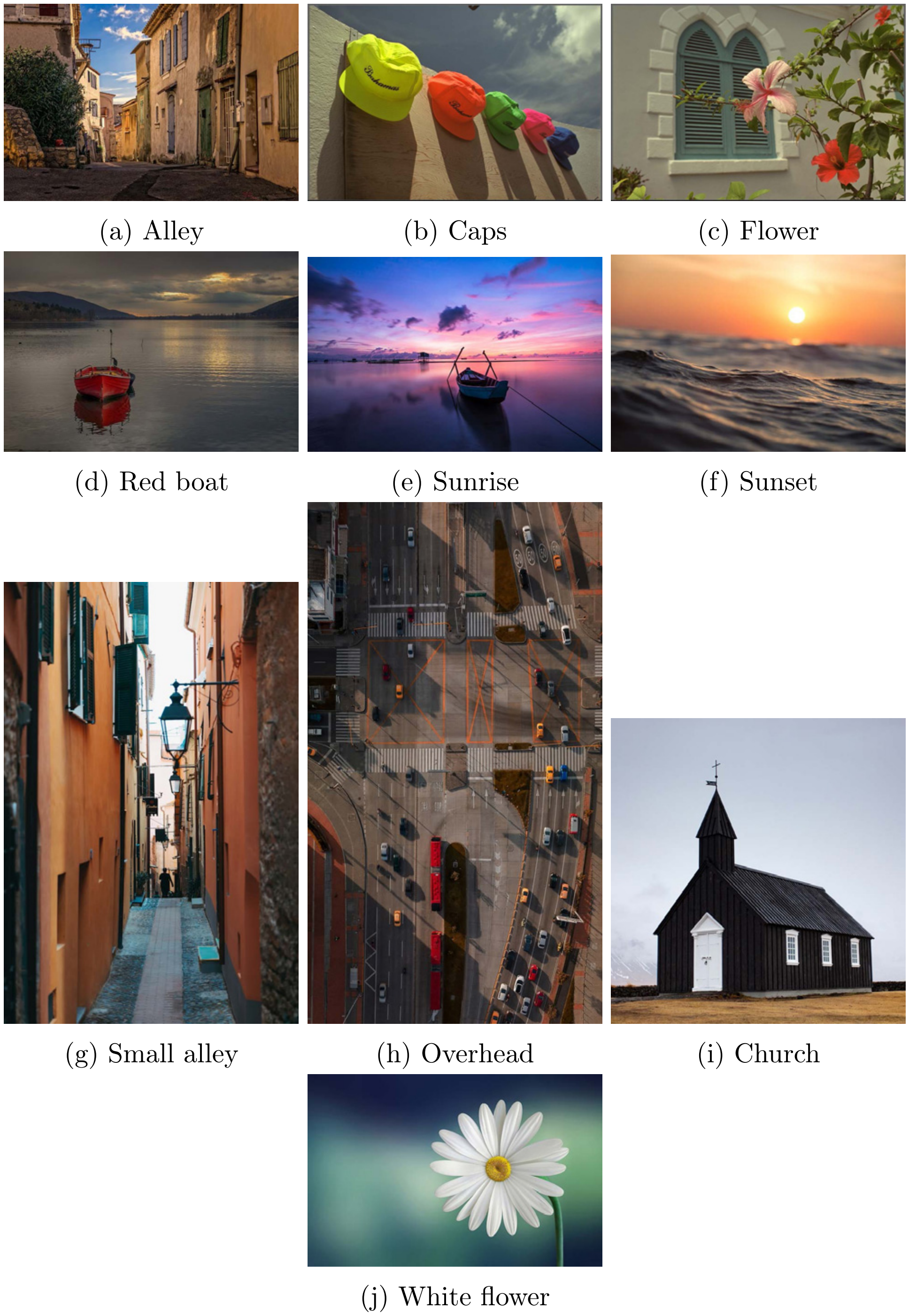


(a) Image with STREAD

(b) Image with STRESS

(c) Original image

Figure 11. Image after STREAD and STRESS.



(a) Alley (b) Caps (c) Flower

(d) Red boat (e) Sunrise (f) Sunset

(g) Small alley (h) Overhead (i) Church

(j) White flower

Figure 12. Image set used in survey.

standard test images with a good range of different properties in terms of level of detail, contrast, and colours. The original images are shown in Figure 12. Seven pairs (STRESS and STREAD) were computed for each original image, with iterations $N \in \{300, 350, 400, 450, 500, 550, 600\}$. All the resulting images are available as supplementary material.

QuickEval [32] was used for setting up and running the experiment in the lab. A gray background was used behind the images and a 200 ms delay was added before going to the next image to avoid the memory effect from the previous stimulus. The experiment was run in a controlled room to make sure that there were no other disturbances during the experiment. The computer screen, an Asus PA32UCG with a resolution of 3840×2160, was calibrated using an i1 Display Pro. The colours for the screen were in sRGB and the luminance 300 cd/m^2. The distance from the participants was 70 cm, with the room fully lit. The dynamic ranges of the

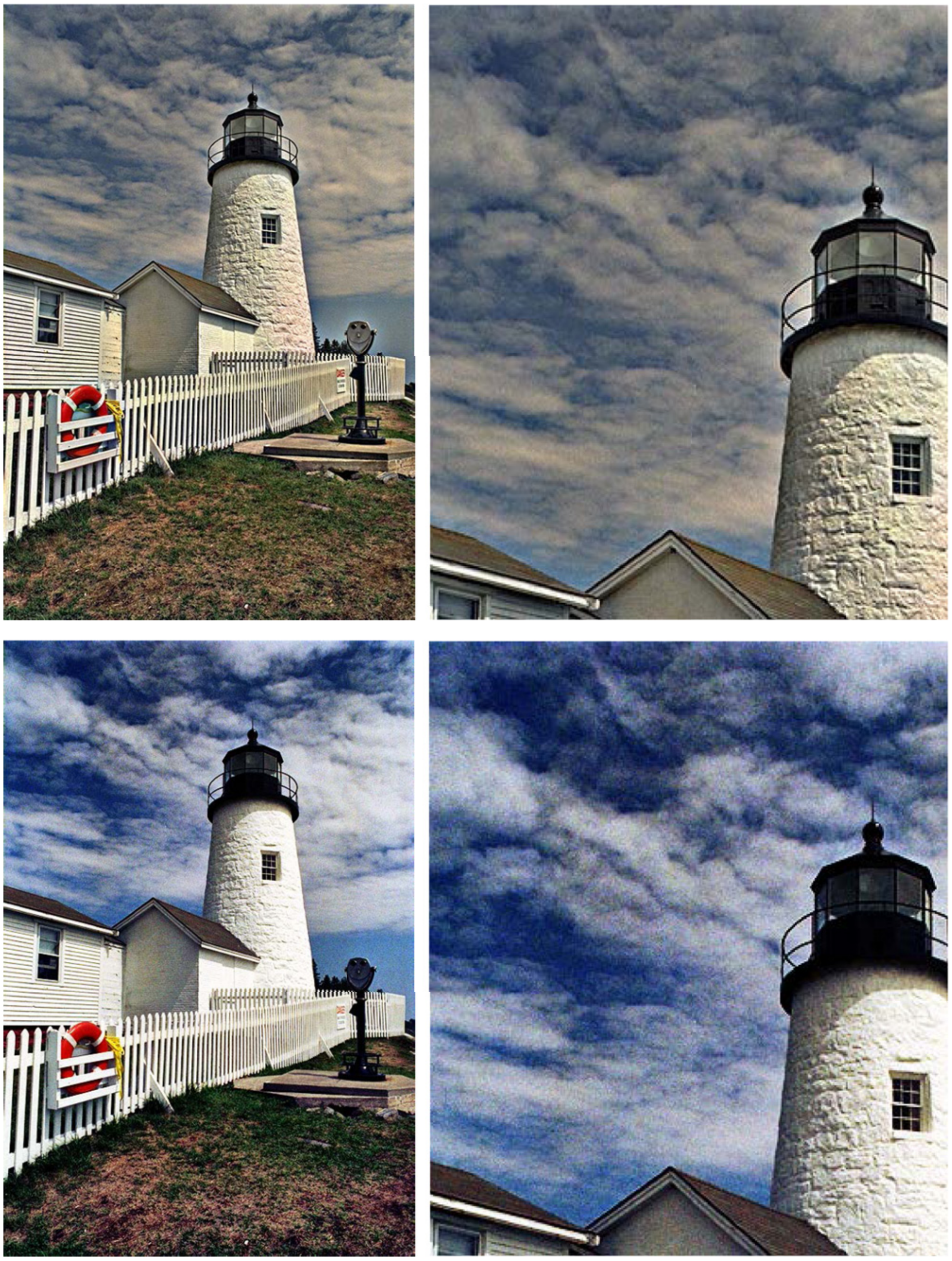


Figure 13. A comparison of the chromatic noise. STREAD above, STRESS below, cropped and zoomed version on the right.

display and the selected images were not so high that glare was a relevant factor.

The experiment was set up on campus and students from the university were picked for the experiment. The students had different academic backgrounds and genders. A total of 22 observers participated in the experiment. The experiment was conducted in two parts. The first part analysed image preference and the second part analysed image noise. For both parts, the observers were given instructions on what to do. For the image noise part, the observers were first shown an image with chromatic noise, Figure 13, to indicate the kind of noise sought. To make sure only results of comparable computational complexity were compared, the participants were shown only the pair of images that had the same number of iterations. So, e.g., STRESS_300 was compared to STREAD_300. The image pairs that were shown had their placement randomized, to avoid bias toward one of the images.

4.2 *Comparing STREAD with the Original Images*

To assess how well STREAD performs in comparison with the original images, we set up a pair-comparison experiment where the 10 original images (Fig. 12) were compared to STREAD with 300, 450 and 600 iterations. This resulted in 30 pairs for observers to evaluate, shown in random order.

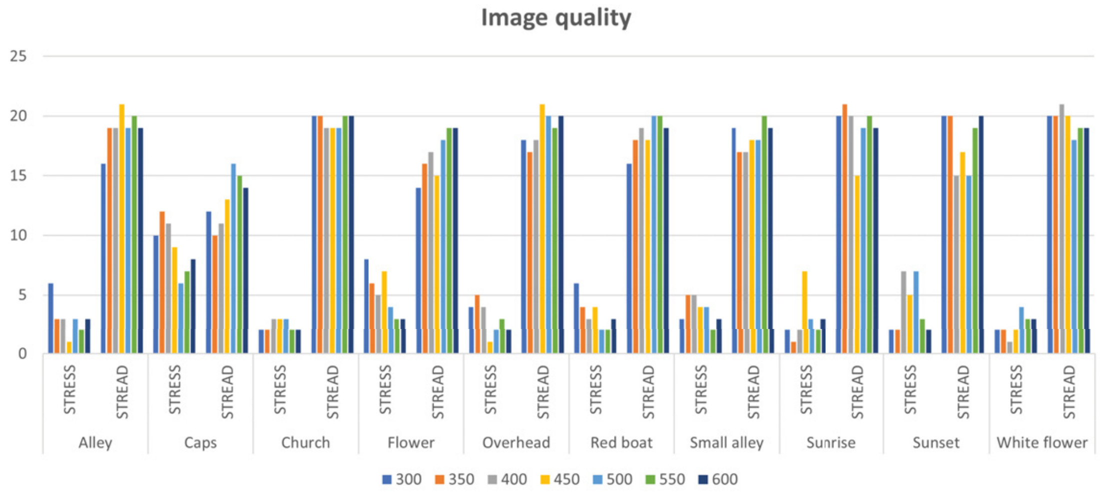


Figure 14. Results from image preference experiment. The bar plot shows the number of times STRESS and STREAD were preferred by the observers for the different images.

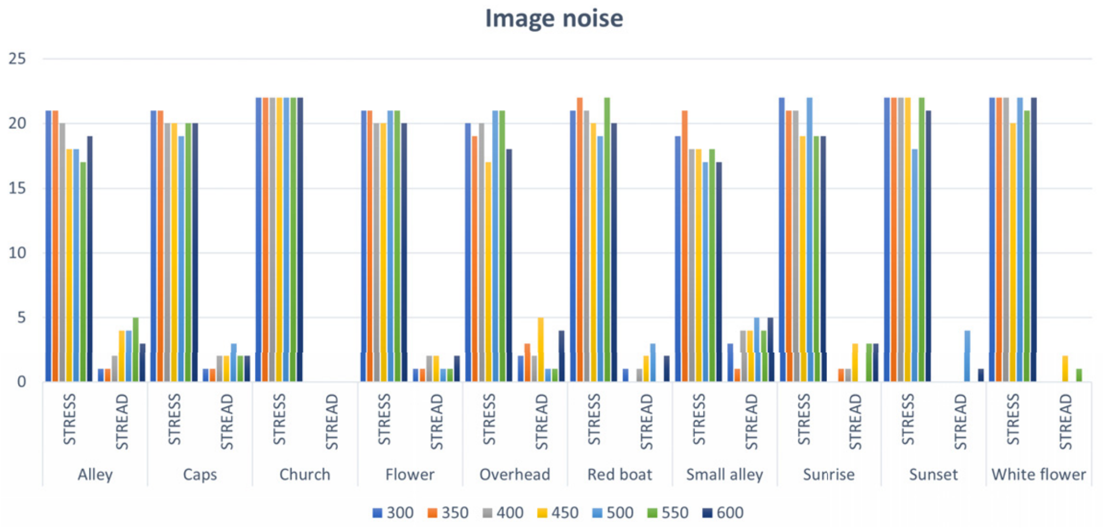


Figure 15. Results from image noise experiment. The bar plot shows the number of times STRESS and STREAD were selected to have higher noise by the observers for the different images.

Observers were asked to select the image they preferred. The experiment was carried out using QuickEval [32], as an uncontrolled online experiment. A total of 31 observers participated in the experiment.

5. RESULTS AND DISCUSSION

5.1 *Subjective Experiments*

The raw data from QuickEval for the image preference part of the subjective experiment is shown in Table I. The table contains the number of times the given image was preferred by the participants. The table is set up such that the image on the y-axis is the one selected. For example, for the Alley image at $N = 300$ iterations, the STRESS image was selected by 6 observers and the stress image STREAD image by 16. The same data is shown graphically in Figure 14.

To analyse the statistical significance of the results from the experiment, a two-sided binomial test with the null hypothesis $H_0 : p = 1/2$ was conducted. The resulting p-values for the individual comparisons are given in Table II. Using a threshold of $p < 0.05$, as shown in the coloured cells of Table II, we can see that many of the results are statistically significant, all in favour of STREAD. Even with thresholds of $p < 0.01$ and $p < 0.001$, many results are still significant. Even though not all the individual comparisons are statistically significant, it should be noted that there is not one single occurrence of a statistically significant preference of STRESS over STREAD.

Combining all iterations for STREAD and STRESS for each image, the binomial test gives the p-values shown in Table III. Here we see that the p-values are so small that we

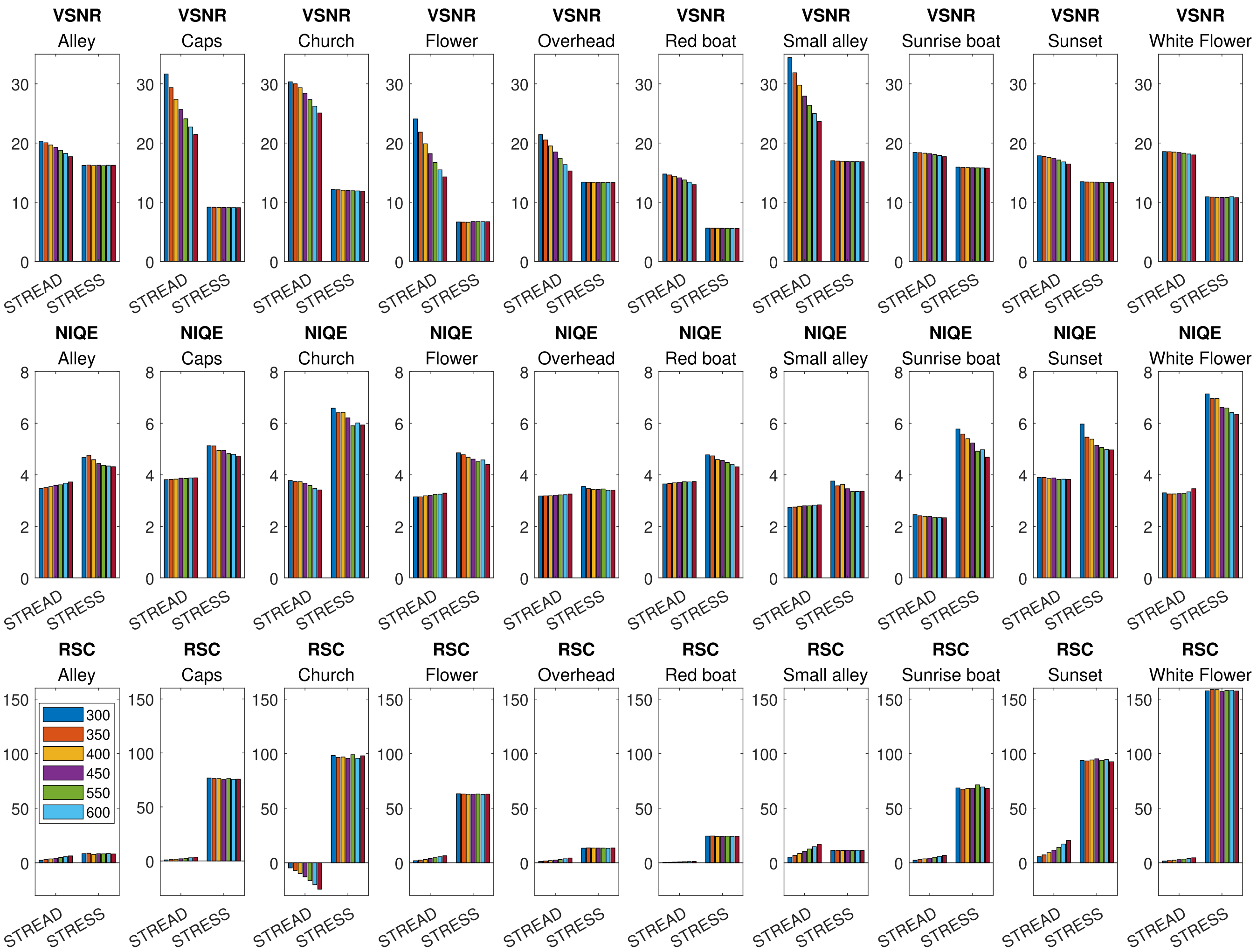
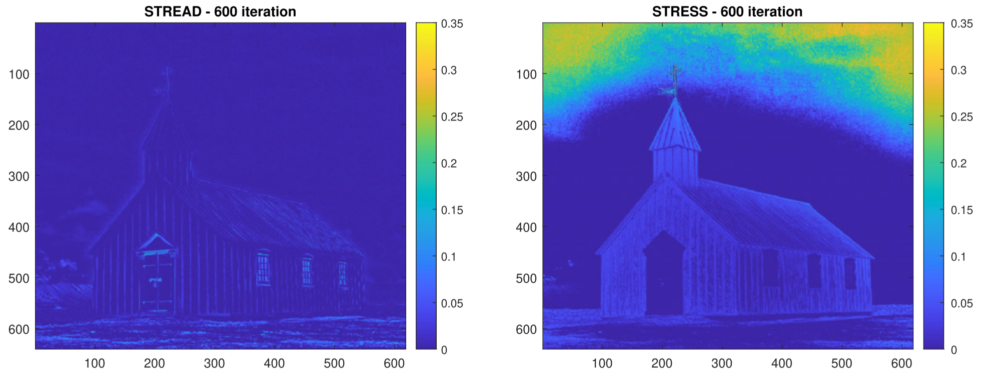


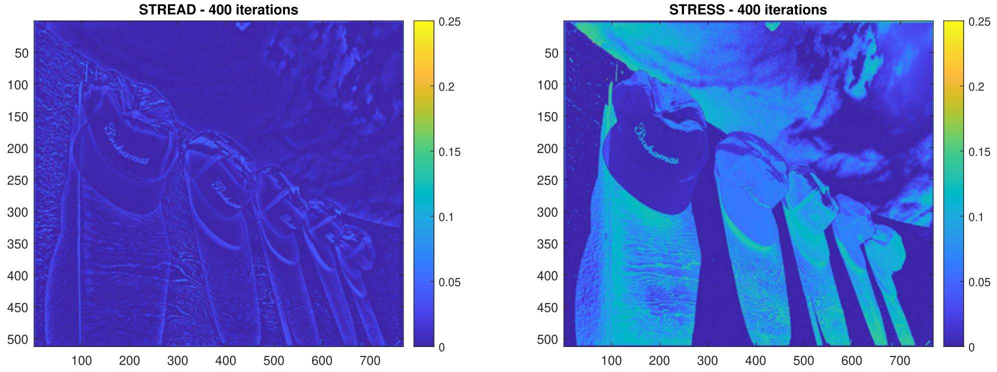
Figure 16. VSNR (top), NIQE (middle) and RSC (bottom) for the 10 images. For VSNR, higher values are better; for NIQE, lower values are better; and for RSC, higher values indicate higher contrast than the original. We can see that for VSNR and NIQE, the proposed STREAD in all images has better values compared to STRESS. For 9 out of the 10 images, STREAD produces images with higher contrast. Colour in the legend indicates the number of iterations.



(a) STREAD

(b) STRESS

Figure 17. The difference between the original and enhanced images for each pixel for the church image, with 600 iterations for each method.



(a) STREAD

(b) STRESS

Figure 18. The difference between the original and enhanced images for each pixel for the caps image, with 400 iterations for each method.

Table I. Raw data for image preference for the individual images (down) and iterations *N* (across).

		300	350	400	450	500	550	600
Alley	STRESS	6	3	3	1	3	2	3
	STREAD	16	19	19	21	19	20	19
Caps	STRESS	10	12	11	9	6	7	8
	STREAD	12	10	11	13	16	15	14
Church	STRESS	2	2	3	3	3	2	2
	STREAD	20	20	19	19	19	20	20
Flower	STRESS	8	6	5	7	4	3	3
	STREAD	14	16	17	15	18	19	19
Overhead	STRESS	4	5	4	1	2	3	2
	STREAD	18	17	18	21	20	19	20
Red boat	STRESS	6	4	3	4	2	2	3
	STREAD	16	18	19	18	20	20	19
Small alley	STRESS	3	5	5	4	4	2	3
	STREAD	19	17	17	18	18	20	19
Sunrise	STRESS	2	1	2	7	3	2	3
	STREAD	20	21	20	15	19	20	19
Sunset	STRESS	2	2	7	5	7	3	2
	STREAD	20	20	15	17	15	19	20
White flower	STRESS	2	2	1	2	4	3	3
	STREAD	20	20	21	20	18	19	19

can easily draw conclusions about which algorithm is best for image preference. Finally, a last binomial test was done on all STREAD versus all STRESS images, resulting in a *p*-value $p = 5.5 \times 10^{-153}$.

The raw data from QuickEval for the subjective experiment on resulting image noise is shown in Table IV. It is set up similar to Table I, and shows the number of times one image was perceived to be more noisy than the other. The corresponding graphical representation is shown in Figure 15. A brief look at the table shows us that the images created with STRESS are selected by far the most, and in some cases, they are the only one selected.

The trend is even more pronounced than for the image preference, which is also confirmed by the individual *p*-values for the binomial tests shown in Table VI. Contrary to the image preference, there is no *p*-value for this test that has a value larger than 0.05. Using a threshold of $p < 0.05$, the coloured cells, we can infer that the result is statistically significant. However, when performing so many statistical tests from the same data, a Bonferroni correction should be applied, leading to a lower *p*-value threshold. Even with stricter thresholds of $p < 0.01$ and $p < 0.001$, the results are still robust. A binomial test was also conducted on all iterations for STREAD and STRESS for each image, and the results are displayed in Table V. Here, we observe that the *p*-values are so small that we can confidently conclude which algorithm is superior for image preference. Finally, a binomial test was carried out on all STREAD versus all STRESS images, which yielded a *p*-value of $p = 8.99 \times 10^{-288}$.

Comparison of STREAD with the original images is shown in Table VII, where the number of times STREAD or the original is preferred is shown. This reveals that overall for the 10 images, STREAD is, according to a binomial test, significantly better than the original ($p = 1.73 \times 10^{-7}$). Conducting the binomial test on 300, 450 and 600 iterations gives a similar conclusion, being statistically significant at $p < 0.05$ for all *p*-values are $p = 0.0005$, $p = 0.0007$, and $p = 0.0354$ for 300, 450 and 600 iterations, respectively. We can notice that with higher iterations, the preference towards the original is increasing slightly. Analysis of the images reveals that in 7 images STREAD has a higher number than the original and lower in 3 images (Alley, Sunrise, and White flower). This can indicate that content plays a role.

5.2 *Objective Image Metrics*

We evaluated the performance of the suggested STREAD and compared it to STRESS using objective metrics. The Visual-Signal-to-Noise-Ratio (VSNR) [33] is a full-reference

Table II. *p*-values for image preference with individual images (across) and iterations *N* (down). White: $p \geq 0.05$, yellow: $0.05 > p \geq 0.01$, green: $0.01 > p \geq 0.001$, blue: $p < 0.001$.

				Images		
		Alley	Caps	Church	Flower	Overhead
	300	5.2×10^{-2}	8.3×10^{-1}	1.2×10^{-4}	2.9×10^{-1}	4.3×10^{-3}
	350	8.6×10^{-4}	8.3×10^{-1}	1.2×10^{-4}	5.2×10^{-2}	1.7×10^{-2}
	400	8.6×10^{-4}	1.0	8.6×10^{-4}	1.7×10^{-2}	4.3×10^{-3}
N	450	1.1×10^{-5}	5.2×10^{-1}	8.6×10^{-4}	1.3×10^{-1}	1.1×10^{-5}
	500	8.6×10^{-4}	5.2×10^{-2}	8.6×10^{-4}	4.3×10^{-3}	1.2×10^{-4}
	550	1.2×10^{-4}	1.3×10^{-1}	1.2×10^{-4}	8.6×10^{-4}	8.6×10^{-4}
	600	8.6×10^{-4}	2.9×10^{-1}	1.2×10^{-4}	8.6×10^{-4}	1.2×10^{-4}
				Images		
		Red boat	Small alley	Sunrise	Sunset	White flower
	300	5.2×10^{-2}	8.6×10^{-4}	1.2×10^{-4}	1.2×10^{-4}	1.2×10^{-4}
	350	4.3×10^{-3}	1.7×10^{-2}	1.1×10^{-5}	1.2×10^{-4}	1.2×10^{-4}
	400	8.6×10^{-4}	1.7×10^{-2}	1.2×10^{-4}	1.3×10^{-1}	1.1×10^{-5}
N	450	4.3×10^{-4}	4.3×10^{-3}	1.3×10^{-1}	1.7×10^{-2}	1.2×10^{-4}
	500	1.2×10^{-4}	4.3×10^{-3}	8.6×10^{-4}	1.3×10^{-1}	4.3×10^{-3}
	550	1.2×10^{-4}	1.2×10^{-4}	1.2×10^{-4}	8.6×10^{-4}	8.6×10^{-4}
	600	8.6×10^{-4}	8.6×10^{-4}	8.6×10^{-4}	1.2×10^{-4}	8.6×10^{-4}

Table III. *p*-values for each image across iterations.

Alley	4.2×10^{-21}
Caps	2.9×10^{-2}
Church	1.7×10^{-24}
Flower	2.2×10^{-11}
Overhead	4.2×10^{-21}
Red boat	8.2×10^{-19}
Small alley	2.2×10^{-17}
Sunrise	6.5×10^{-22}
Sunset	4.8×10^{-16}
White flower	1.7×10^{-24}

metric for the detection of distortions in natural images; this is done by using contrast thresholds and visual masking to determine if the distortion is visible. VSNR is therefore a good metric to compare visible distortions in STREAD and STRESS. Default parameters for VSNR are used, these being Alpha $= 0.04$ and viewing parameters equal to $b = 0$, $k = 0.02874, g = 2.2, r = 138, v = 27.5$, num_levels $= 5$ and filter_gains = 2.ˆ (1:num_levels) The second metric is the no-reference Natural Image Quality Evaluator (NIQE) [34], which is a no-reference metric based on analysis of statistical features from natural scene statistics and correlates well with subjective scores on various distortions, including noise [35]. For NIQE, we calculate the results for each colour channel and average them. The last objective metric is a contrast metric, the RSC [36] which is a weighted multilevel contrast metric. For RSC, we have used the optimal parameters from Simone et al. [36]. We calculate the difference in contrast between the contrast enhanced images and the original image, so a higher value will indicate increased contrast and a lower value indicates decreased contrast compared to the original.

The results for VSNR, NIQE, and RSC are shown in Figure 16. Higher VSNR values are better, given in dB in the range 0 - Inf, while for NIQE, lower values are better, where it has been found that values higher than 40 rarely occur [37]. We see that for VSNR, all values for STREAD are higher than STRESS, while for NIQE, all values for STREAD are lower than STRESS. For all images, VSNR indicates that STREAD has less visible distortions compared to STRESS. It can also be seen that the difference between STREAD and STRESS is image-dependent, and consistent with the results from the subjective experiment. For the contrast metric RSC, we notice that increasing the number of iterations produces images with higher contrast compared to the original. The exception is the image Church, where the STREAD algorithm produces very uniform areas without noise, which lowers the local contrast and therefore leads to a lower RSC value. It can also be seen that STRESS produces higher RSC values, which is partly due to the global contrast change because of the colour shift in STRESS.

To supplement the analysis, we have also calculated the difference between the original and the enhanced images. This has been done by taking the normalized sum of the absolute difference between the original and enhanced images for each pixel for both STREAD and STRESS. We visualize the results for the church image with 600 iterations in Figure 17. It can be seen that the higher differences that

Table IV. Raw data for image noise for the individual images (down) and iterations N (across).

		300	350	400	450	500	550	600
Alley	STRESS	21	21	20	18	18	17	19
	STREAD	1	1	2	4	4	5	3
Caps	STRESS	21	21	20	20	19	20	20
	STREAD	1	1	2	2	3	2	2
Church	STRESS	22	22	22	22	22	22	22
	STREAD	0	0	0	0	0	0	0
Flower	STRESS	21	21	20	20	21	21	20
	STREAD	1	1	2	2	1	1	2
Overhead	STRESS	20	19	20	17	21	21	18
	STREAD	2	3	2	5	1	1	4
Red boat	STRESS	21	22	21	20	19	22	20
	STREAD	1	0	1	2	3	0	2
Small alley	STRESS	19	21	18	18	17	18	17
	STREAD	3	1	4	4	5	4	5
Sunrise	STRESS	22	21	21	19	22	19	19
	STREAD	0	1	1	3	0	3	3
Sunset	STRESS	22	22	22	22	18	22	21
	STREAD	0	0	0	0	4	0	1
White flower	STRESS	22	22	22	20	22	21	22
	STREAD	0	0	0	2	0	1	0

STRESS produces in the sky, the noise increases. The result for the caps image is shown for 400 iterations in Figure 18. We can notice the same observation regarding noise in the sky for STRESS, and that STREAD, in general, makes fewer changes

Table V. p-values for image preference with individual images (across) and iterations N (down). White: $p \geq 0.05$, yellow: $0.05 > p \geq 0.01$, green: $0.01 > p \geq 0.001$, blue: $p < 0.001$.

		Images				
		Alley	Caps	Church	Flower	Overhead
	300	1.1×10^{-5}	1.1×10^{-5}	4.8×10^{-7}	1.1×10^{-5}	1.2×10^{-4}
	350	1.1×10^{-5}	1.1×10^{-5}	4.8×10^{-7}	1.1×10^{-5}	8.6×10^{-4}
	400	1.2×10^{-4}	1.2×10^{-4}	4.8×10^{-7}	1.2×10^{-4}	1.2×10^{-4}
N	450	4.3×10^{-3}	1.2×10^{-4}	4.8×10^{-7}	1.2×10^{-4}	1.7×10^{-2}
	500	4.3×10^{-3}	8.6×10^{-4}	4.8×10^{-7}	1.1×10^{-5}	1.1×10^{-5}
	550	1.7×10^{-2}	1.2×10^{-4}	4.8×10^{-7}	1.1×10^{-5}	1.1×10^{-5}
	600	8.6×10^{-4}	1.2×10^{-4}	4.8×10^{-7}	1.2×10^{-4}	4.3×10^{-3}
				Images		
		Red boat	Small alley	Sunrise	Sunset	White flower
	300	1.1×10^{-5}	8.6×10^{-4}	4.8×10^{-7}	4.8×10^{-7}	4.8×10^{-7}
	350	4.8×10^{-7}	1.1×10^{-5}	1.1×10^{-5}	4.8×10^{-7}	4.8×10^{-7}
	400	1.1×10^{-5}	4.3×10^{-3}	1.1×10^{-5}	4.8×10^{-7}	4.8×10^{-7}
N	450	1.2×10^{-4}	4.3×10^{-3}	4.8×10^{-7}	4.8×10^{-7}	1.2×10^{-4}
	500	8.6×10^{-4}	1.7×10^{-2}	4.8×10^{-7}	4.3×10^{-3}	4.8×10^{-7}
	550	4.8×10^{-7}	4.3×10^{-3}	8.6×10^{-4}	4.8×10^{-7}	1.1×10^{-5}
	600	1.2×10^{-4}	1.7×10^{-2}	8.6×10^{-4}	1.1×10^{-5}	4.8×10^{-7}

Table VI. *p*-values for each image across iterations.

Alley	6.5×10^{-22}
Caps	2.5×10^{-28}
Church	8.8×10^{-47}
Flower	1.4×10^{-31}
Overhead	1.3×10^{-23}
Red boat	9.9×10^{-33}
Small alley	2.2×10^{-17}
Sunrise	1.9×10^{-30}
Sunset	6.1×10^{-38}
White flower	5.3×10^{-41}

Table VII. Raw data for image preference for the individual images (down) and iterations *N* (across) for STREAD compared to the original. A total of 31 observers participated in the experiment.

		300	450	600	Sum
Alley	Original	21	22	23	66
	STREAD	10	9	8	27
Caps	Original	13	11	11	45
	STREAD	18	20	20	58
Church	Original	12	9	11	32
	STREAD	19	22	20	61
Flower	Original	10	9	11	30
	STREAD	21	22	20	63
Overhead	Original	8	12	12	32
	STREAD	23	19	19	61
Red boat	Original	9	9	9	27
	STREAD	22	22	22	66
Small alley	Original	9	7	8	24
	STREAD	22	24	23	69
Sunrise	Original	14	18	15	47
	STREAD	17	13	16	46
Sunset	Original	9	8	11	28
	STREAD	22	23	20	65
White flower	Original	19	20	25	64
	STREAD	12	11	6	29
Sum	Original	124	125	136	385
	STREAD	186	185	174	545

to the images and treats the image more locally. STRESS increases the edge of the shadows of the caps, while STREAD can also enhance the areas between the shadows.

6. CONCLUSION

We have proposed an alternative algorithm for computing the envelopes of the STRESS algorithm using linear anisotropic diffusion, leading to the STREAD algorithm. The main goal of the transition was to reduce the chromatic noise and thus increase the overall image preference. Both subjective experiments and objective image metrics show that both of these goals were achieved with the new approach, and that the STREAD images were preferred over both STRESS images and the originals with statistical significance.

ACKNOWLEDGMENT

This research was funded by the Research Council of Norway over the project “Individualised Colour Vision-based Image Optimisation”, Grant No. 287209.

REFERENCES

[1] E. H. Land, “The retinex,” *American Scientist* (Wiley, Hoboken, NJ, 1965), Vol. 52, pp. 247–264.

[2] E. H. Land and J. J. Mccann, “Lightness and retinex theory,” J. Opt. Soc. Am. **61**, 1–11 (1971).

[3] E. H. Land, “The Retinex Theory of Color Vision,” Sci. Am. **237**, 108–129 (1977) ISSN: 00368733, 19467087 http://www.jstor.org/stable/24953876 (visited on 22/05/2023).

[4] J. J. McCann and A. Rizzi, *The Art and Science of HDR Imaging* (John Wiley & Sons, New York, 2011).

[5] A. S. Parihar and K. Singh, “A study on Retinex based method for image enhancement,” *2018 2nd Int’l. Conf. on Inventive Systems and Control (ICISC)* (IEEE, Piscataway, NJ, 2018), pp. 619–624.

[6] A. S. Baslamisli, H.-A. Le, and T. Gevers, “CNN based learning using reflection and retinex models for intrinsic image decomposition,” *Proc. IEEE Conf. on Computer Vision and Pattern Recognition* (IEEE, Piscataway, NJ, 2018), pp. 6674–6683.

[7] C. Wei, W. Wang, W. Yang, and J. Liu, “Deep retinex decomposition for low-light enhancement,” *British Machine Vision Conf. 2018* (2018).

[8] J. Liang, Y. Xu, Y. Quan, J. Wang, H. Ling, and H. Ji, “Deep bilateral retinex for low-light image enhancement,” Preprint, arXiv:2007.02018 (2020).

[9] Ø. Kolås, I. Farup, and A. Rizzi, “Spatio-temporal retinex-inspired envelope with stochastic sampling: A framework for spatial color algorithms,” J. Imaging Sci. Technol. **55**, 040503-1–040503-10 (2011).

[10] E. Provenzi, M. Fierro, A. Rizzi, L. D. Carli, D. Gadia, and D. Marini, “Random spray retinex: A new retinex implementation to investigate the local properties of the model,” IEEE Trans. Image Process. **16**, 162–171 (2007).

[11] G. Simone and I. Farup, “Spatio-temporal retinex-like envelope with total variation,” *6th European Conf. on Colour in Graphics, Imaging, and Vision (CGIV)* (Amsterdam, The Netherlands, 2012), pp. 176–181.

[12] G. Gianini, A. Rizzi, and E. Damiani, “A retinex model based on absorbing markov chains,” Inf. Sci. **327**, 149–174 (2016).

[13] J.-B. Thomas and I. Farup, “Demosaicing of periodic and random color filter arrays by linear anisotropic diffusion,” J. Imaging Sci. Technol. **62**, 50401-1 (2018).

[14] I. Farup, “Individualised halo-free gradient-domain colour image daltonisation,” J. Imaging **6**, 116 (2020).

[15] J. A. Frankle and J. J. McCann, “Method and apparatus for lightness imaging,” US Patent 4,384,336 (1983).

[16] D. Marini and A. Rizzi, “A computational approach to color adaptation effects,” Image Vis. Comput. **18**, 1005–1014 (2000).

[17] A. Rizzi, D. Marini, and L. D. Carli, “LUT and multilevel Brownian Retinex colour correction,” Mach. Graphics Vis. Int. J. **11**, 153–168 (2002).

[18] J. J. McCann, “Capturing a black cat in shade: Past and present of Retinex color appearance models,” J. Electron. Imaging **13**, 36–47 (2004).

[19] T. J. Cooper and F. A. Baqai, “Analysis and extensions of the Frankle-McCann Retinex algorithm,” J. Electron. Imaging **13**, 85–92 (2004).

[20] G. D. Finlayson, S. D. Hordley, and M. S. Drew, “Removing shadows from images,” *Proc. 7th European Conf. on Computer Vision* (Springer, Berlin, 2002), pp. 823–836.

[21] R. Montagna and G. D. Finlayson, “Constrained pseudo-Brownian motion and its application to image enhancement,” J. Opt. Soc. Am. A **28**, 1677–1688 (2011).

[22] D. J. Jobson, Z.-u. Rahman, and G. A. Woodell, “Properties and performance of a center/surround retinex,” IEEE Trans. Image Process. **6**, 451–462 (1997).

[23] K. Barnard and B. Funt, “Investigations into multi-scale retinex,” *Proc. Colour Imaging in Multimedia’98* (1998), pp. 9–17.

[24] F. O. Huck, C. L. Fales, R. E. Davis, and R. Alter-Gartenberg, “Visual communication with retinex coding,” Appl. Opt. **39**, 1711–1730 (2000).

[25] J. D. Cowan and P. C. Bressloff, “Visual cortex and the Retinex algorithm,” Proc. SPIE **4662**, 278–285 (2002).

[26] Z.-u. Rahman, D. J. Jobson, and G. A. Woodell, “Retinex processing for automatic image enhancement,” J. Electron. imaging **13**, 100–110 (2004).

[27] P. Perona and J. Malik, “Scale-space and edge detection using anisotropic diffusion,” IEEE Trans. Pattern Anal. Mach. Intell. **12**, 629–639 (1990).

[28] D. Tschumperle and R. Deriche, “Vector-valued image regularization with PDEs: A common framework for different applications,” IEEE Trans. Pattern Anal. Mach. Intell. **27**, 506–517 (2005).

[29] S. Di Zenzo, “A note on the gradient of a multi-image,” Comput. Vis. Graph. Image Process. **33**, 116–125 (1986) https://www.sciencedirect.com/science/article/pii/0734189X86902239.

[30] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, “High-resolution image synthesis with latent diffusion models,” *2022 IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)* (IEEE, Piscataway, NJ, 2022).

[31] B. Xia, Y. Zhang, S. Wang, Y. Wang, X. Wu, Y. Tian, W. Yang, and L. V. Gool, “DiffIR: Efficient diffusion model for image restoration,” *Proc. IEEE/CVF Int’l. Conf. on Computer Vision (ICCV)* (2023).

[32] K. V. Ngo, C. A. Dokkeberg, I. Farup, and M. Pedersen, “Quickeval: A web application for psychometric scaling experiments,” Proc. SPIE **9396**, 212–224 (2015).

[33] D. M. Chandler and S. S. Hemami, “VSNR: A wavelet-based visual signal-to-noise ratio for natural images,” IEEE Trans. Image Process. **16**, 2284–2298 (2007).

[34] N. Venkatanath, D. Praneeth, M. C. Bh, S. S. Channappayya, and S. S. Medasani, “Blind image quality evaluation using perception based features,” *2015 21st National Conf. on Communications (NCC)* (IEEE, Piscataway, NJ, 2015), pp. 1–6.

[35] A. Mittal, R. Soundararajan, and A. C. Bovik, “Making a “completely blind” image quality analyzer,” *IEEE Signal Processing Letters* (IEEE, Piscataway, NJ, 2012), Vol. 20, pp. 209–212.

[36] G. Simone, M. Pedersen, and J. Y. Hardeberg, “Measuring perceptual contrast in digital images,” J. Vis. Commun. Image Represent. **23**, 491–506 (2012).

[37] A. Zvezdakova, D. Kulikov, D. Kondranin, and D. Vatolin, “Barriers towards no-reference metrics application to compressed video quality analysis: On the example of no-reference metric NIQE,” *29th Int’l. Conf. on Computer Graphics and Vision, CEUR Workshop Proc.* (RWTH, Aachen, 2019), pp. 22–27.