Master's thesis

NTNU
Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of Computer Science

Petter Sagvold

Spatio-Temporal Retinex-Inspired Envelope with Anisotropic Diffusion

Master's thesis in Applied Computer Science
Supervisor: Ivar Farup
May 2023

NTNU
Norwegian University of
Science and Technology

Petter Sagvold

Spatio-Temporal Retinex-Inspired Envelope with Anisotropic Diffusion

Master's thesis in Applied Computer Science
Supervisor: Ivar Farup
May 2023

Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of Computer Science

NTNU
Norwegian University of
Science and Technology

Spatio-Temporal Retinex-Inspired Envelope with Anisotropic Diffusion

Petter Sagvold

May 30, 2023

Abstract

In the past, several algorithms have been suggested for spatial color correction of digital images. One of these algorithms is STRESS, a Retinex inspired algorithm that uses stochastic sampling to obtain the envelope function. This sampling method creates some chromatic noise on low iterations, meaning more iterations are needed. In this project, we present a new method using anisotropic diffusion instead of stochastic sampling that we call STREAD, Spatio-Temporal Retinex-Inspired Envelope with Anisotropic Diffusion. Anisotropic diffusion is implemented for the sampling and used for contrast enhancement, and this new implementation is shown to create better images than with STRESS. An experiment with observers are done for both image quality and image noise, to evaluate the two approaches. The results show that STREAD is significant better in both image quality and the image noise generated, as judged by the observers. The next steps for the new implementation are to test it for other applications like spatial color gamut mapping and local color correction of images.

Keywords

Image enhancement techniques, Retinex, anisotropic diffusion, STRESS, spatial color algorithm

Sammendrag

Flere algoritmer har tidligere blitt foreslått for spatiell fargekorrigering av bilder. En av disse algoritmene er STRESS, en Retinex-inspirert algortime som bruker stokastisk sampling for å lage omhyllingskurver. Denne samplingmetoden skaper en del støy ved lave iterasjoner, noe som betyr at flere iterasjoner er nødvendig for et godt resultat. I dette prosjektet presenterer vi en ny metode som bruker anisotropisk diffusjon i stedet for stokastisk sampling som vi kaller STREAD. Anisotropisk diffusjon er implementert for sampling og er brukt for kontrastforsterkning, og denne nye implementasjonen er vist at skaper bedre bilder enn det STRESS gjør. Et eksperiment med observatører er gjennomført for både bildekvalitet og bildestøy, for å evaluere de to algoritmene. Resultatene viser at STREAD er betydelig bedre på både bildekvalitet og bildestøy, bedømt av observatørene. De neste trinnene for denne implementasjonen er å teste den for andre applikasjoner, som spatiell fargeomfangstilpassning og lokal fargekorrigering av bilder.

Acknowledgements

I would like to express my sincere appreciation and gratitude to my supervisor Ivar Farup for his guidance and academic feedback throughout the entire process of this master thesis. Your knowledge and feedback has been very helpful for this project.

I would also like to extend my sincere thanks to Marius Pedersen for his expert guidance on the experiment side of the research project. Your provided knowledge and feedback was really valuable for the project.

Contents

Figures

Tables

Chapter 1

Introduction

1.1 Topic

When trying to capture an image with a device like a camera, any kind of disturbance can lead to a degradation of the image quality [1]. And when dealing with computer vision, images are the most critical aspect of the field. Most of the vision-based techniques, like object tracking, object detection and human activity recognition, require the image or video to be of a good quality to accomplish their task [2]. So good quality images are required in vision-based methods. To make sure that these images have a good quality, there are some techniques present that can help improve the quality. Such techniques are called image enhancement techniques [3]. The basic purpose of image enhancement is to improve the quality of an image for human eyesight. Many different techniques have appeared in recent time, with varying effectiveness and efficiency. One of these techniques is called Retinex, and this technique can be used to serve the purpose of effective enhancement of the low-quality image [1, 4]. Retinex is a combination of the words Retina and Cortex and is based on this complete scenario that how a viewpoint is perceived by the human visual system.

The human visual system is a complex system that includes both the eyes and the brain [5]. When light hits the retina, a light-sensitive layer of tissue at the back of the eye, special cells called photoreceptors turn the incoming light into electrical signals [6, 7]. These electrical signals then travel from the retina through the optic nerve to the brain. The brain then turns these signals into the images we see.

When trying to capture an image with the help of a camera or another device, there is a possibility that due to some conditions, the resulting image will end up with a low dynamic range or having poor color constancy. In terms of image, the ratio of the highest pixel value and the lowest pixel value is known as Dynamic Range of that image. In the human visual system, we perceive the color of an object as constant with various illumination conditions, this is called as color constancy [1]. These problems are some of the things techniques like Retinex try to fix. Another of these image enhancement methods is STRESS, which is an algorithm based on Retinex. The problem STRESS have is that the resulting images contain some chromatic noise. To fix this, the algorithm has to iterate for a longer period and average the result to get rid of the noise. This comes down to their sampling method. By implementing another sampling method, like anisotropic diffusion, we should be able to get the noise level down or gone completely. This then leads us into the goals and research questions for the project.

1.2 Goals and Research Questions

The aim of this research project is to implement another sampling method for envelopes in STRESS. Stochastic sampling used in STRESS creates chromatic noise when running with a low number of iterations. This then forces the algorithm to run for more iterations to get rid of the image noise. By using anisotropic diffusion for the sampling, we should be able to minimize the image noise, since anisotropic diffusion is a technique aiming at reducing image noise. In order to have a clearer grasp of the project, and to more precisely specify what is to be done, a group of research questions has been made.

Research question 1: *Can anisotropic diffusion be used for the implementation of envelopes for STRESS?*

Research question 2: *Does the new implementation create images with better quality than STRESS?*

Research question 3: *Does the new implementation create images with less image noise than STRESS?*

1.3 Contribution

This research study will first and most importantly make a significant contribution to the area of research by expanding and adding a new implementation for spatial color and Retinex based algorithms. The new sampling method implemented, the technology used, evaluation and results, and all other parts of this research study presents a new way to create spatial color algorithms. This research study has, as a result of this, the potential to assist researchers in this field.

1.4 Thesis Structure

Chapter 2: Background and theory An explanation of the human color vision theory called Retinex, some previous work around that topic and other theory related work and research that is relevant for this thesis.

Chapter 3: STREAD An overview and explanation on how this implementation is different from the other and how it works.

Chapter 4: Experiments and Results The experimental setup and how it was conducted.

Chapter 5: Results Results from the implementation and the results from the experiment.

Chapter 6: Discussion An evaluation of the results from the experiments and the discussion around the given results

Chapter 7: Conclusion A conclusion of the results and discussion, conclusion of the whole research and a look into the future work of the thesis and the topic.

Chapter 2

Background and Theory

2.1 Retinex

A great amount of research has been done on Human Visual System, HVS, which is quite difficult to mimic as the visual system has many complex and robust mechanisms to acquire useful information from the physical environment [8]. In particular, the color of an area in a visual scene is heavily influenced by the chromatic content of the other areas of the scene [9]. This psychophysiological phenomenon is known as locality of color perception [8].

Retinex is one of the earliest models that can deal with the locality of perception, and the model was proposed by Land and McCann in 1971 [10]. The Retinex model is an image processing method that exhibits some behaviors that are similar to the human visual system [9]. Edwin Land's Retinex hypothesis of human color vision was proposed to explain color perceptions in actual scenes. Research on color constancy has shown that color does not correspond to receptor responses [11]. In actual scenes, appearances are controlled by the overall image's material. Any color can be seen in an L, M, and S cone response triplet. The authors came up with the term "Retinex" to describe the spatial image processing that underlies color constancy [4, 12, 13]. The term Retinex is a contraction of the words retina (human eye) and cortex (human mind). Additionally, they demonstrated how three lightnesses seen in long-, middle-, and short-wave illumination predict color perception [9].

The Retinex model and its various applications has since the introduction kept the scientific community interested [14, 15]. By using long paths to scan across the images, locality is achieved in the basic implementation of Retinex. After the Retinex model was introduced, there were many different implementations and analysis that quickly followed. Each of the different implementation and analysis of the model differ in how they each achieve locality [9]. Some of the implementation explore the images by using paths or computing ratios with neighbors in a multilevel framework [16–21] or using Brownian motions models [22, 23]. Other implementations computes values over the given image with convolution mask or weighting distances [24–28].

2.1.1 Milano-Retinex

Milano-Retinex is a spatial color algorithm grounded on the Retinex theory and widely applied to enhance the visual content of real-world color images. Milano-Retinex and Retinex share the same core, but the different ways of implementing it results in very important differences [29]. Milano-Retinex algorithms are usually applied as image enhancers, with no pre- or post-filtering calibration, and take a digital color image as input and output a color enhanced image [30]. In [31], they replaced the old path-based scanning with a new approach using random sampling composed by a cloud of points. They called the cloud of points a spray and was arranged around the pixel to do the computation. The reason for this new approach was to avoid the redundancy of the Brownian path sampling [30, 32]. This new approach was then called Random Spray Retinex, or RSR for short. Random Spray Retinex takes an RGB image as the input and processes the color channels separately [30, 31]. Under is a list of Milano-Retinex algorithms that either derive and/or are inspired by this new method.

- Light-RSR [33]
- swRSR [34]
- QBRIX [35]
- RSR-P [36]
- STRESS [37]
- STRETV [8]
- T-Rex [38]
- GREAT [39]
- GRASS [40]

Out of these algorithms, we will take a closer look at both STRESS and STRETV. The reason why is that the new implementation proposed in this project will be based on STRESS and will have similarities with STRETV.

2.1.2 STRESS

STRESS, Spatio-Temporal Retinex-Inspired Envelope with Stochastic Sampling, introduced in [37] is an algorithm that tries to reproduce some of the adjustment mechanisms typical for the Human Visual System (HVS). In this article, the authors present a new method to characterize the local visual context using two envelopes E^{max} and E^{min}. The envelopes are then constructed in a way so that the original image signal is between these two envelopes. E^{max} and E^{min} are calculated using stochastic sampling, and with a simple weighting of the values obtain an edge-preserving method. Even with the framework being called Retinex-Inspired, it cannot be considered a Retinex implementation, since it implements pixel value stretching that is very different [37].

The authors also present how STRESS can be used in other possible applications. The most straightforward application that STRESS can be used in, is local contrast enhancement of grayscale images. Since the envelopes can be interpreted as local reference maximum and minimum points, to obtain a local effect, the pixel value should be compared to these quantities [37]. The same approach used in contrast enhancement can also be used for color correction of color images, where the calculation is performed independently on the channels and the envelopes defines the local maximum and minimum. Other implementation mentioned is HDR image rendering, spatial color gamut mapping, temporal color correction of movies and color to grayscale conversion [37].

But the algorithm is not without some problems. When doing a low number of iterations, STRESS has trouble with chromatic noise appearing in the images. To reduce this, the sampling process is iterated several times and then averaged. By doing this, the noise level is strongly decreased, but in turn the time of computation is increased.

2.1.3 STRETV

One way of solving the sampling and chromatic noise problem in STRESS, is by changing the stochastic sampling with another sampling method. In STRETV, Spatio-Temporal Retinex-like Envelope with Total Variation [8], they introduce an algorithm that instead of using the sampling in STRESS uses the total variation method to calculate the envelopes. This new implementation shows promising results when used in contrast enhancement and in automatic color correction. The algorithm has a lower computational complexity compared to STRESS, $O(N * n)$ compared to $O(N * M * n)$, and also performs better when it comes to the resulting images containing less chromatic noise.

2.2 Anisotropic Diffusion

Anisotropic diffusion is a technique used in computer vision and image processing that aims to reduce the image noise without removing significant parts of the image [41, 42]. This is typically edges, lines or other detail that are important for the interpretation of the image [43, 44]. By utilizing a constant diffusion coefficient, the anisotropic diffusion equations reduce to the heat equations, which is equivalent to Gaussian blurring [45]. Anisotropic diffusion is also sometimes called Perona-Malik diffusion, since the formulation proposed in the article [41] was referred to as anisotropic. The formulation was referred to as anisotropic, but was in fact isotropic. Our implementation will be based on the Perona-Malik diffusion, but our will be local, linearized and anisotropic. Anisotropic diffusion can be used for enhancement of ultrasound images [46], speckle reduction [47, 48], enhancement of 3-D angiogram [49] and for hyperspectral imagery enhancement [50]. Despite the effectiveness of anisotropic diffusion in image denoising, many studies [51, 52] exists to analyze its impact in other image processing techniques, like compression and inpainting.

Chapter 3

STREAD

We introduce a new model for spatio-temporal image enhancement, which we call 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 propose to use anisotropic diffusion [53]. The equations that describe the STREAD model are given below.

3.1 Proposed Method

The components of the structure tensor S of the original image can be expressed as shown in Equation (3.1) [54].

$$S_{ij} = \sum_{\mu,\nu} \frac{\partial u_0^{\mu}}{\partial x^i} \frac{\partial u_0^{\nu}}{\partial x^j} \tag{3.1}$$

The eigenvalues of the structure tensor S are denoted λ^+ and λ^-, and the corresponding normalized eigenvectors e^+ and e^- are stored as columns in the orthonormal eigenvector matrix E, such that the structure tensor can be written $S = E^T \operatorname{diag}(\lambda^+, \lambda^-)E$ [55]. From this, the diffusion tensor is then defined as Equation (3.2)

$$\mathbf{D} = E^T \operatorname{diag}(d(\lambda^+), d(\lambda^-)), E \tag{3.2}$$

where $d(\lambda)$ is a nonlinear diffusion coefficient function Equation (3.3) [55] whose task is to suppress the diffusion across the edges while preserving it along the edges,

$$d(\lambda) = \frac{1}{1 + \kappa\lambda^2} \tag{3.3}$$

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

This together with the diffusion equations in [56] and [55] gives us the anisotropic diffusion equations Equation (3.4) for e_{max} and Equation (3.5) for e_{min}. These equations are almost identical, but the difference between them is that Equation (3.4) calculates the e_{max} envelope and Equation (3.5) calculates the e_{min} envelope.

$$\frac{\partial e_{max}}{\partial t} = \nabla(\mathbf{D}\nabla e_{max}) - \lambda(e_{max} - u_0) \quad \text{s.t.} \quad e_{max} \geq u_0 \tag{3.4}$$

$$\frac{\partial e_{min}}{\partial t} = \nabla(\mathbf{D}\nabla e_{min}) - \lambda(e_{min} - u_0) \quad \text{s.t.} \quad e_{min} \leq u_0 \tag{3.5}$$

A data attachment is added to both Equation (3.4) and Equation (3.5) so that both the calculations don't lead to entirely uniform envelopes, which then leads to global behavior of the resulting algorithm. 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. For each iteration, e_{max} and e_{min} are cut so that they are always over or under the original image. To avoid that there are values closely outside the $[0, 1]$ interval, values outside this interval for e_{max} and e_{min} are cut. The envelopes and the original image are then put into Equation (3.6) [37], where p_0 is the original image, e_{max} and e_{min} the envelopes, and p the contrast enhanced image.

$$p = \frac{p_0 - e_{min}}{e_{max} - e_{min}} \tag{3.6}$$

3.2 Resulting Envelopes

As we have seen before, the main feature of both STRESS and STREAD algorithms is the computation of the envelopes e_{max} and e_{min} for each channel of the image. These envelopes can be applied in various image enhancement methods as explained in Section 2.1.2. In Figure 3.1, we can see how the envelopes for STRESS and STREAD look like, and how they are used to increase the contrast of the original image. The envelopes also contain the original image content, always brighter (e_{max}) or darker (e_{min}), and preserve the edges of the image.

3.3 Impact of Parameters

As with the STRESS algorithm, the behavior of STREAD can also vary according to the values of its parameters. We will now explore in more detail how these parameters influence the output of the algorithm.

3.3.1 Iterations

Although both converge to the final solution, iterations in STRESS and STREAD do quite different things. STRESS iterates several times and averages the process

(a) e_{max} STRESS, 1000 iterations **(b)** e_{min} STRESS, 1000 iterations

(c) e_{max} STREAD, 1000 iterations **(d)** e_{min} STREAD, 1000 iterations

Figure 3.1: Envelopes: $\kappa = 1000$ for STREAD, sampling points $= 3$ for STRESS

to reduce the sampling chromatic noise, while iterations in STREAD brings you closer to the solution from your starting point. This starting point can either be from the original image or from the top/bottom. The initial tests for STREAD were done with iterations $= [10, 20, 50, 100]$ just to see how it worked. It did not show a lot of difference in the enhanced image, but the envelopes changed a bit for each iteration. In Figure 3.2 you can see the envelopes for STREAD with 100 iterations.

After some more small tests on iterations, a bigger test was conducted using iterations $= [100, 200, 300, 400, 500, 600, 700, 800, 900, 1000]$. With this test, it was easier to see the difference in what higher iterations did with the output. In Figure 3.3 and Figure 3.4 you can see the difference between 100 iterations and 1000, as well as the envelopes used. The edges are much sharper in the image with the most iterations, but you can also see some artifacts starting to appear near edges.

3.3.2 Kappa

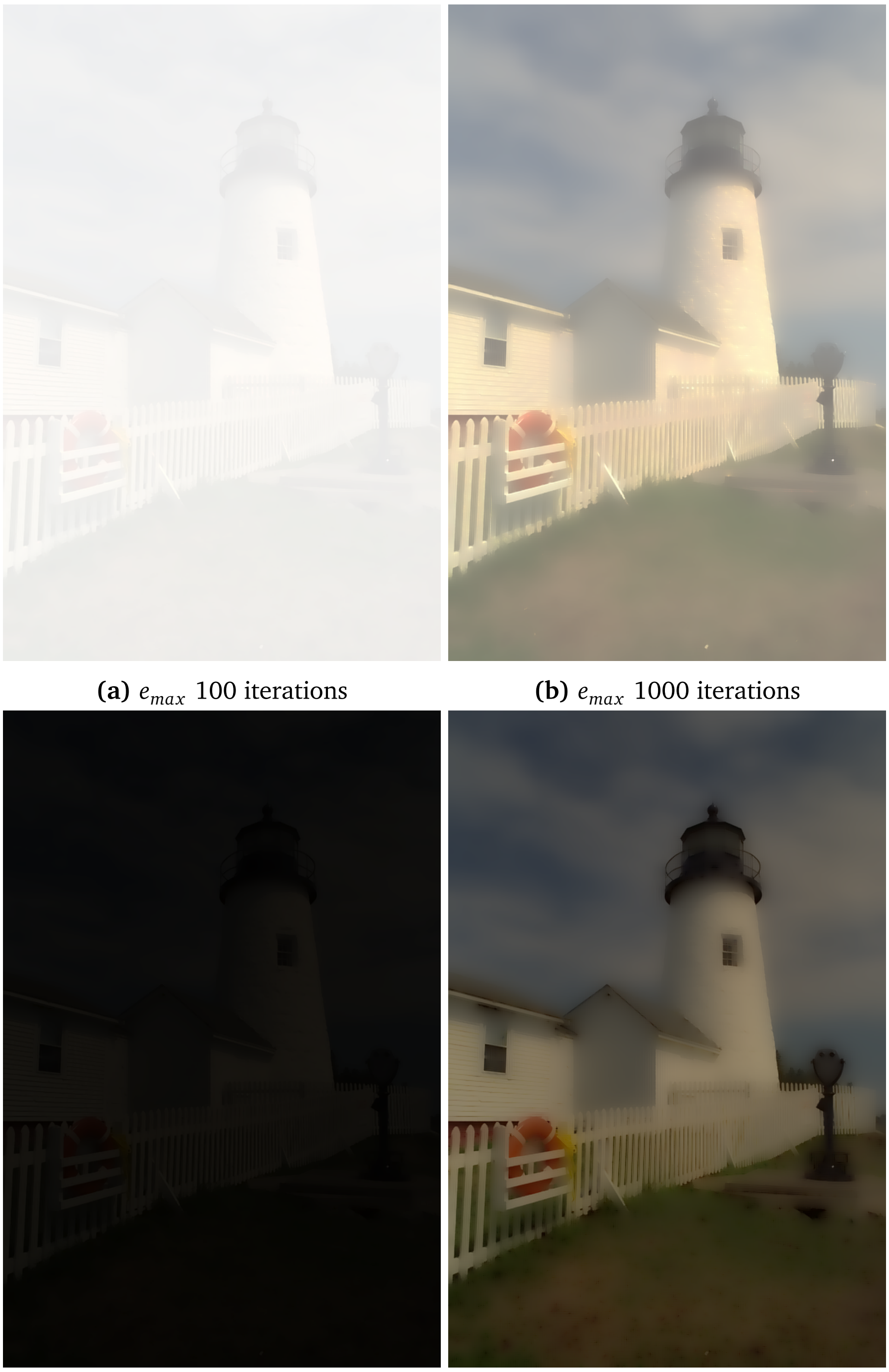
As with iterations, tests were done with kappa to figure out what value it had to be to give the best output. While testing the highest value for iterations, a value of $\kappa = 1000$ was used. When the iterations got higher than 700, artifacts like halos started to appear on edges in the image. In Figure 3.5 you can see how the halos appear around edges in the image.

(a) e_{max} STRESS **(b)** e_{min} STRESS

Figure 3.2: e_{max} and e_{min} with 100 iterations

(a) 100 iterations **(b)** 1000 iterations

Figure 3.3: STREAD: 100/1000 iterations, $\kappa = 1000$



(a) e_{max} 100 iterations **(b)** e_{max} 1000 iterations

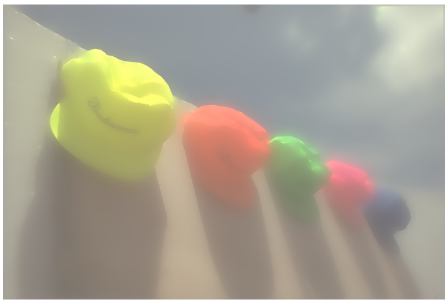
(c) e_{min} 100 iterations **(d)** e_{min} 1000 iterations

Figure 3.4: STREAD envelopes: 100/1000 iterations, $\kappa = 1000$

Figure 3.5: Image with artifact

(a) e_{max}: iter = 1000, $\kappa = 1000$

(b) e_{max}: iter = 1000, $\kappa = 100000$



(c) e_{min}: iter = 1000, $\kappa = 1000$

(d) e_{min}: iter = 1000, $\kappa = 100000$

Figure 3.6: Envelopes $\kappa = 1000$ and $\kappa = 100000$

(a) $\kappa = 1000$

(b) $\kappa = 5000$

(c) $\kappa = 10000$

(d) $\kappa = 50000$

(e) $\kappa = 100000$

Figure 3.7: Images with different values of κ

To counter this, a higher value of κ was needed, since higher values of $kappa$ will give more edge preservation in the image. This can be seen in Figure 3.6. Values for the parameters used for this test were iterations $= 1000$ and $\kappa = [1000, 5000, 10000, 50000, 100000]$ and the outputs are shown in Figure 3.7. Even with $\kappa = 100000$ and iterations $= 1000$, some artifacts still appear around some edges, but lower iterations did not have artifacts. Knowing this, κ was set to 10000 and iterations $= [300, 350, 400, 450, 500, 550, 600]$ since these values did not have any artifacts but still had good outputs.

3.3.3 Lambda

The last variable is λ, which is used for the data attachment. The data attachment term $-\lambda(u - u_0)$ in Equation (3.4) and Equation (3.5) is a regularization term

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

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

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

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

Figure 3.8: $\lambda = 0.01/0.001$

that incorporates the prior information of the original image into the enhancement process, and acts as a constraint that minimizes the discrepancy between the envelope and the original image [57, 58]. If $\lambda = 0$, the envelopes will be flat, which result in the algorithm becoming spatially independent or global. If $\lambda -> \infty$, the envelopes will be the same as the original image in the end. So finding an optimal value for λ i important. In the first test runs, the value for λ was set to 0.01. This quickly gave bad results when the iterations got higher than 200. These results can be seen in Figure 3.8. λ was then changed to 0.001 and the results got much better.

Chapter 4

Experiment

The quality of the images generated by STREAD and STRESS was compared by a psychometric survey to address RQ2 and RQ3. Initial tests showed that STREAD performed well with $\kappa = 10000$, $\lambda = 0.001$ and iterations ranging from 300 to 600. A simple script was used to produce images with both STRESS and STREAD. A sample of 10 images was selected for a paired comparison evaluation. These images are shown in figure 4.4 and 4.5. The script made 7 pairs for each image, with iterations [300, 350, 400, 450, 500, 550, 600].

One "problem" that the first run of the script showed us was the difference in color for STREAD and STRESS. While STREAD kept the same colors as the original, STRESS made the images a bit brighter and a bit blue. See figure 4.1 STRESS was modified a bit to fix this. Two linear scaling for preserving white and gray and a gamma correction for gray were added. The difference between these two can be seen in figure 4.2. The linear scaling to preserve gray was chosen since it was the ones that looked the closest to the original and STREAD. Even with the color correction, some images came out so bad that it was no reason to use them. Figure 4.3 is one of them. The images were then replaced with other images that performed better and were of better quality. This was also done to not create bias for the participants toward one of the algorithms.

(a) Image with STREAD

(b) Image with STRESS

(c) Original image

Figure 4.1: Image after STREAD and STRESS

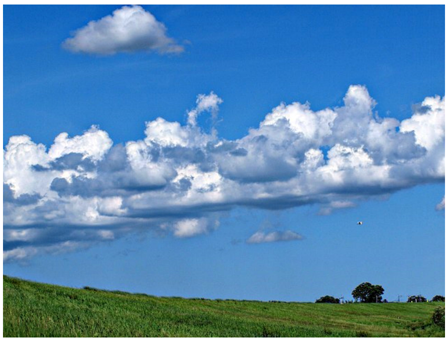
(a) Default STRESS

(b) Linear scaling to preserve white

(c) linear scaling to preserve gray

(d) Gamma correction to preserve gray

Figure 4.2: Color corrected images for STRESS



(a) STREAD

(b) STRESS

Figure 4.3: Image not used for experiment

4.1 Setup

QuickEval(**cite**) was used for setting up and running the experiment. A gray background was used behind the images and a 200-millisecond delay was added when going to the next image. Having the delay reduces the memory effect from the previous stimuli. The survey was run in a controlled room to make sure that there were no other disturbances during the experiment. The computer screen was calibrated using an i1 Display Pro, colors for the screen were in sRGB and the luminance 300 cd/m^2. A screenshot of how the survey looked can be seen in figure 4.6.

4.2 Running the Experiment

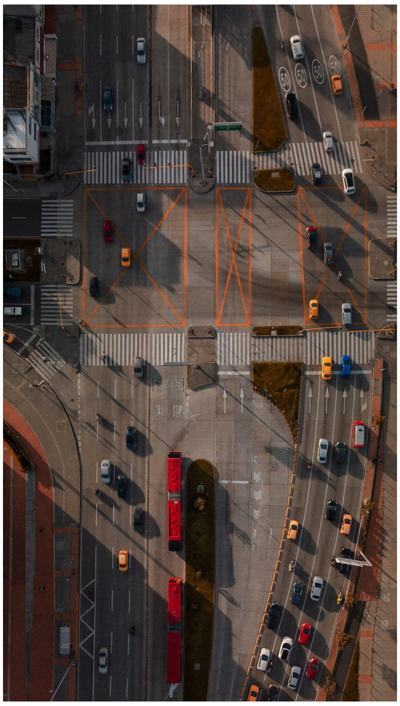
The experiment was set up on campus and a group of students from the university was picked to do the experiment. The students had different academic backgrounds and genders. This is quite a narrow group because of the location of the experiment, which is something to keep in mind when analyzing the results. The experiment was done in two part, one part about image quality and the other part for image noise. For both parts they were presented with instructions on what to do, which is shown in Figure 4.7. For the image noise part, they were also shown a noisy image, Figure 4.8, so everyone knew what kind of noise we were looking for. The participants were shown only the pair of images which had the same iteration number, so STRESS_300 was compared to STREAD_300. The image pairs that was shown had their placement randomized, to avoid some of the bias toward one of the images. The image pairs was not showed twice, which could have decreased more bias but also would have increased the time to finish the experiment.

(a) Alley

(b) Caps

(c) Church

(d) Flower



(e) Overhead

Figure 4.4: Image set used in survey

(a) Red boat

(b) Sunrise

(c) Sunset

(d) White flower

(e) Small alley

Figure 4.5: Image set used in survey

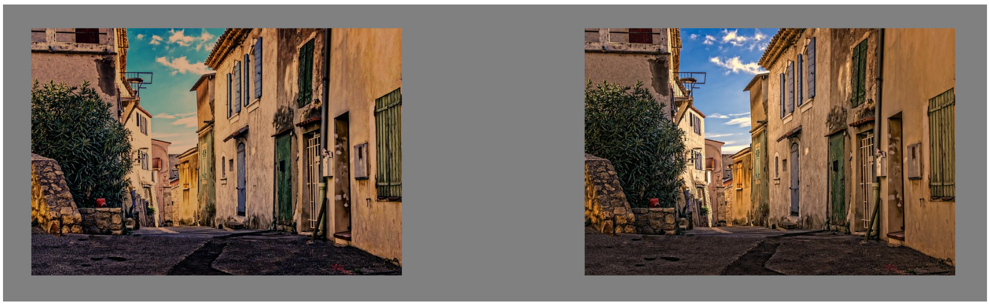
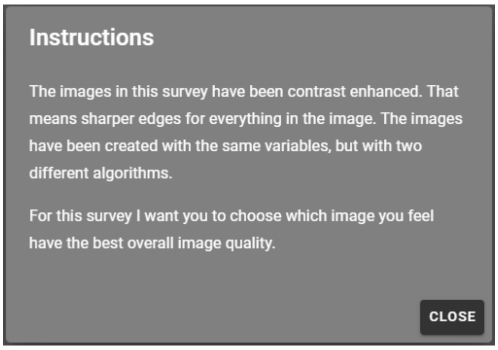


Figure 4.6: Example of comparison in QuickEval



(a) Instructions for image quality

(b) Instructions for image noise

Figure 4.7: Instructions for survey

Figure 4.8: Image with noise

Chapter 5

Results

This chapter presents the results of the data collection and analysis conducted for this research. The purpose of this research was to investigate STREAD in comparison with STRESS. The research questions were:

- RQ1: Can anisotropic diffusion be used for the implementation of envelopes for STRESS?
- RQ2: Does the new implementation create images with better quality than STRESS?
- RQ3: Does the new implementation create images with less image noise than STRESS?

RQ1 has already been answered in Chapter 3 and Chapter 4, so we will answer RQ2 and RQ3 in this chapter. The data for this research was collected through the experiment mentioned earlier in Chapter 4, and was analyzed using descriptive and inferential statistics, including binomial tests.

The main results of the data analysis are:

- The image quality for STREAD is better than STRESS
- STREAD creates little image noise in the images
- There are some interesting differences in the images used when it comes to image quality
- Good image quality is a bit different between participants

The following sections will report and present the results in more detail, organized by experiment and research question. Each section will provide a summary of the analysis method, the results in text and graphical form, comments from the participants, and an interpretation of the given result. The chapter will conclude with a summary of the main findings.

5.1 Image Quality

This section reports the results of the image quality experiment that was conducted to compare the image quality between images created with STREAD and alley_300 (2)

	STREAD_iter_300_kappa_10000_lambda_0.001.png	STRESS_iter_300_sampling_3.png
STREAD_iter_300_kappa_10000_lambda_0.001.png		16
STRESS_iter_300_sampling_3.png	6	

Figure 5.1: Table for data from QuickEval

STRESS. How the experiment was executed is described in Chapter 4. The experiment aimed to answer the following research question:

- RQ2: Does the new implementation create images with better quality than STRESS?

The data from the experiment was recorded in QuickEval for each image pair. As mentioned earlier, each image had 7 paired comparisons, which gave us 70 small tables combined into one. Figure 5.1 show how these small tables are displayed. The table contains the data on what image the participants thought had the best image quality. The table is set up so that the image on the y-axis is the image picked. That means in this table, STREAD was picked 16 times and STRESS 6. All these "smaller" tables were then combined into Table 5.2 to make it easier to analyze the results. Comments made by the participants were also written down, with some of them being:

- "Both images look nice"
- "It's hard to choose the best image"
- "I like the color on this image, but this one have better edges"
- "It is hard to choose which one has the best quality, because they both have different things which is good"
- "So much information on the image"
- "This one looks much nicer, too much noise in the sky"
- "This was harder than I imagined"

Table 5.2 is presented with the iterations used on the x-axis, and the images with STRESS and STREAD on the y-axis. So as an example, *Alley* with 500 iterations was selected 3 times for STRESS and 19 times for STREAD. So higher numbers mean more people selected that image. This table is the basis for all the plots, both for the whole table but also for parts of it, and the values are also used later for the binomial tests. Plotting Table 5.1 as a bar chart gives us Figure 5.2. Here, each iteration for STREAD and STRESS is presented as a bar. This also makes it easier to see if STREAD outperforms STRESS when it comes to image quality. But we can also see that some images don't have a clear advantage with either STREAD or STRESS. We have picked out some of the images and data to take a closer look at. These are *Alley*, *Caps* and *Church*.

In both *Alley*, Figure 5.3, and *Church*, Figure 5.5, we can quickly see that STREAD is the preferred choice when it comes to image quality. But if we take a closer look at Figure 5.4 for *Caps*, we can see that they are more close than the

		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

Table 5.1: Raw data for image quality

Figure 5.2: Image quality

Figure 5.3: Image quality: Alley

other images. For the first iterations, 300-400, we can see that STRESS has the same or higher score than STREAD. But with the higher iterations, STREAD have a higher number than STRESS.

To check that the results from the experiment had statistical significance, a binomial test was done with the values in Table 5.2. The p-values from this test were put into Table 5.2 and each of the cells were given a color depending on the value:

- white: $p > 0.05$
- yellow: $0.05 > p > 0.01$
- green: $0.01 > p > 0.001$
- blue: $p < 0.001$

A quick look at the table shows us that we can not draw any conclusion based on the values from *Caps*. This is something we also knew from the plot of the original values for it. Using a threshold of $p < 0.05$, the colored cells, we can see the result is statistical significant. Even with thresholds of $p < 0.01$ and $p < 0.001$, the results are still good. Combining all iterations for STREAD and STRESS for each image, and then doing a binomial test, gives us the results shown in Table 5.3. Here we see that the p-values are so small that we can easily draw conclusions for which algorithm is best for image quality. At the end, a last binomial test was done on all STREAD and all STRESS, which resulted in a p-value $= 5.5 \times 10^{-153}$.

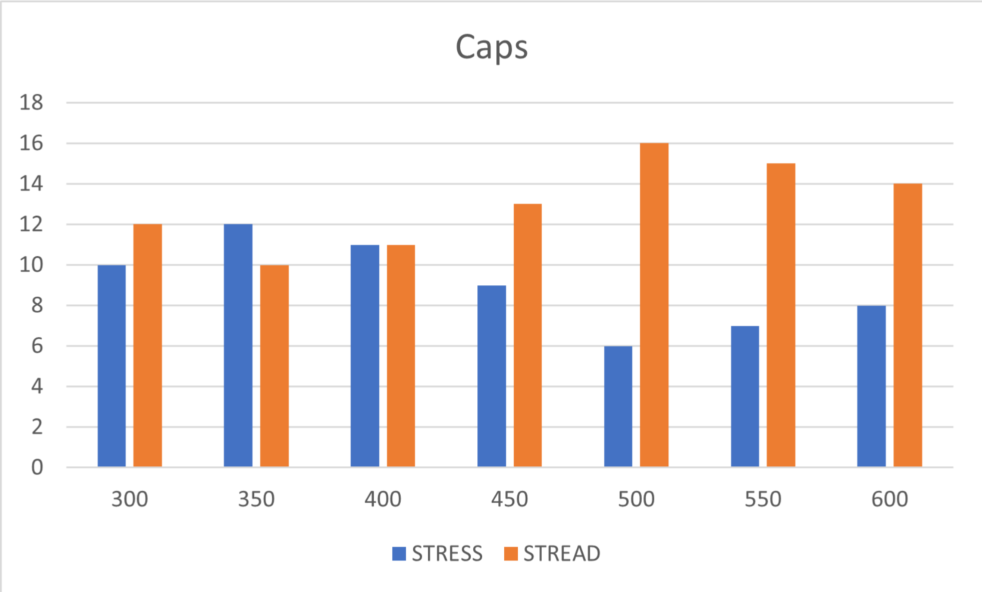


Figure 5.4: Image quality: Caps

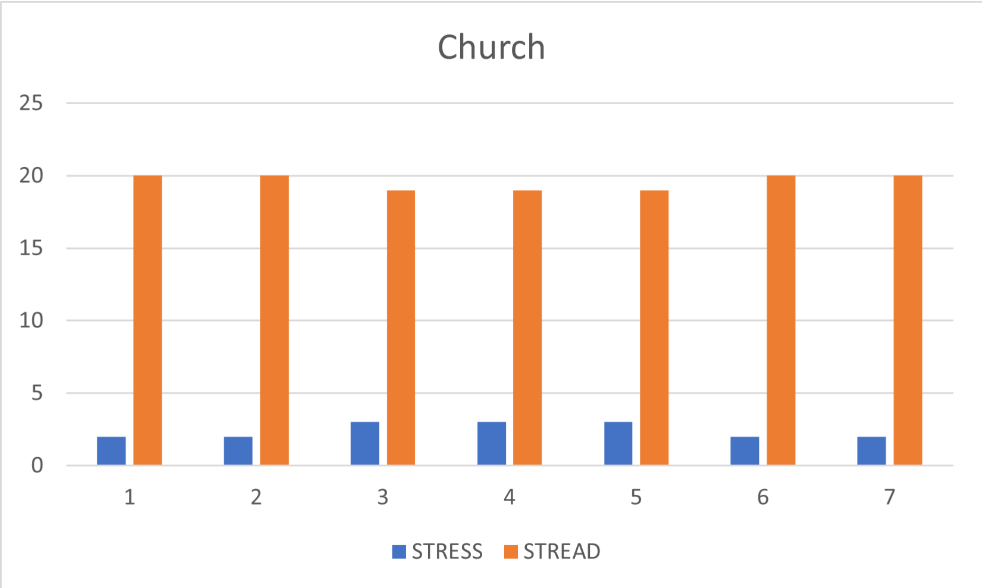


Figure 5.5: Image quality: Church

	Alley	Caps	Church	Flower	Overhead
300	0,052	0,83	0,00012	0,29	0,0043
350	0,00086	0,83	0,00012	0,052	0,017
400	0,00086	1	0,00086	0,017	0,0043
450	0,000011	0,52	0,00086	0,13	0,000011
500	0,00086	0,052	0,00086	0,0043	0,00012
550	0,00012	0,13	0,00012	0,00086	0,00086
600	0,00086	0,29	0,00012	0,00086	0,00012

	Red boat	Small alley	Sunrise	Sunset	White flower
300	0,052	0,00086	0,00012	0,00012	0,00012
350	0,0043	0,017	0,000011	0,00012	0,00012
400	0,00086	0,017	0,00012	0,13	0,000011
450	0,0043	0,0043	0,13	0,017	0,00012
500	0,00012	0,0043	0,00086	0,13	0,0043
550	0,00012	0,00012	0,00012	0,00086	0,00086
600	0,00086	0,00086	0,00086	0,00012	0,00086

Table 5.2: p-values image quality

Alley	4.2×10^{-21}
Caps	0.029
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}

Table 5.3: p-values each image

		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

Table 5.4: Raw data for image noise

5.2 Image Noise

This section reports the results of the image noise experiment that was conducted to compare the image noise generated with STREAD and STRESS. For this experiment, the participants were asked to pick the image that had the most image noise. The experiment aimed to answer the following research question:

- RQ3: Does the new implementation create images with less image noise than STRESS?

All the data from this experiment is put into Table 5.4. It has the same setup as the image quality one, with iterations on the x-axis and the image with STREAD or STRESS on the y-axis. All this data is then put into a plot which we can see in Figure 5.6.

A quick look at the table shows us that the images created with STRESS are

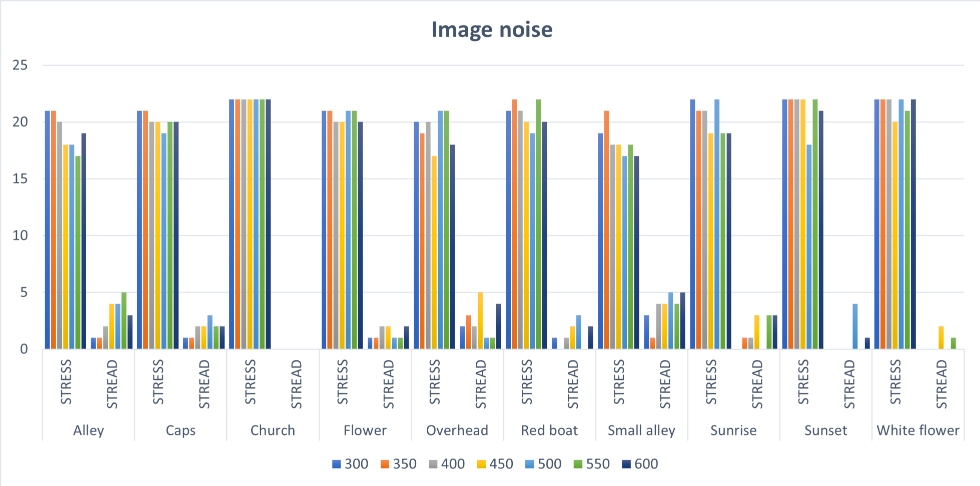


Figure 5.6: Image noise

picked the most, and sometimes the only one picked. Plotting Table 5.4 as a bar chart gives us Figure 5.6. This plot has the same setup as Figure 5.2, where each bar is for each iteration. We then pick out *Alley*, *Caps* and *Church* to take a closer look at their respective plots.

A comparison of Figure 5.7 for *Alley* and Figure 5.8 for *Caps* reveals that STRESS produces the most image noise among the algorithms. However, we can also observe that STREAD is sometimes preferred by some participants, which suggests that there may be some variation in how they define and perceive image noise. For the third image, *Church*, the results are more clear-cut. As shown in Figure 5.9, only STRESS is selected as the algorithm with the most image noise. This can also be seen in the images used in the experiment, which are displayed in Figure 5.10. By inspecting these images closely, we can see that the sky in the image produced by STRESS has a lot of noise that is absent in the image produced by STREAD. This makes STRESS stand out as the noisiest algorithm for this category.

A binomial test was also performed on the values in Table 5.5 to assess the statistical significance of the results from the experiment. The p-values from this test were recorded in Table 5.5 and each of the cells were assigned a color based on the value:

- white: p > 0.05
- yellow: 0.05 > p>0.01
- green: 0.01 > p > 0.001
- blue: p < 0.001

As opposed to the image quality, there is no p-value for this test that has a value larger than 0.05. Using a threshold of $p < 0.05$, the colored cells, we can infer that the result is statistically significant. 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

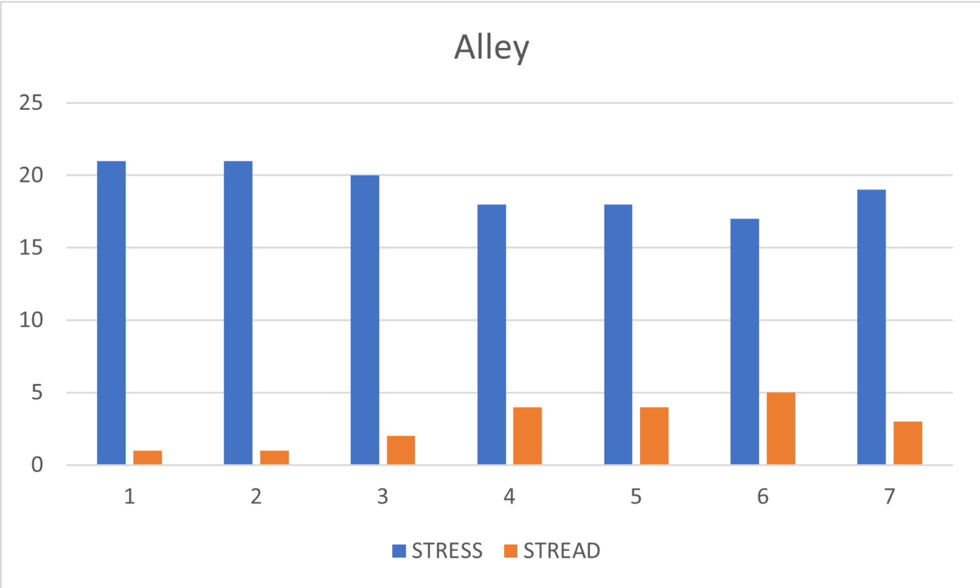


Figure 5.7: Image noise: Alley

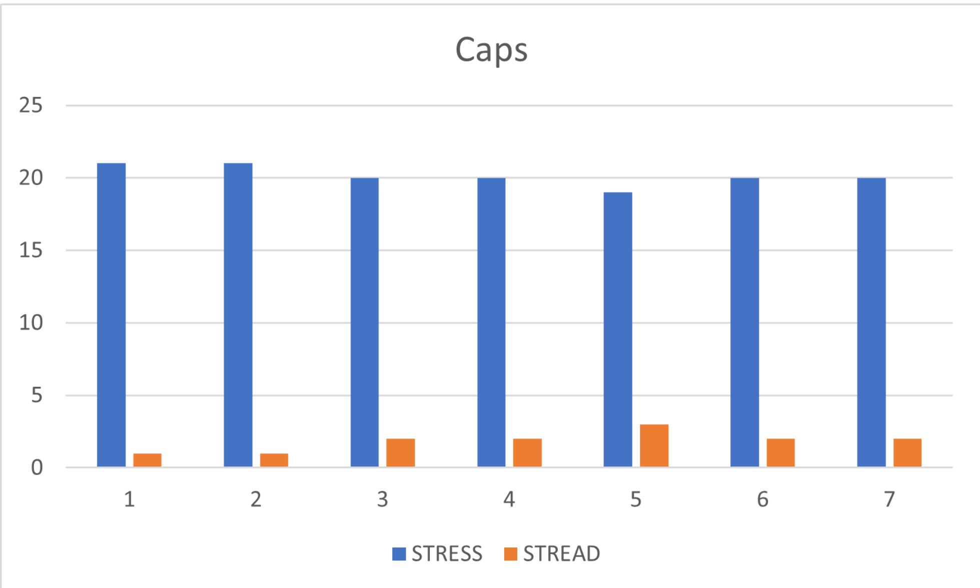


Figure 5.8: Image noise: Caps

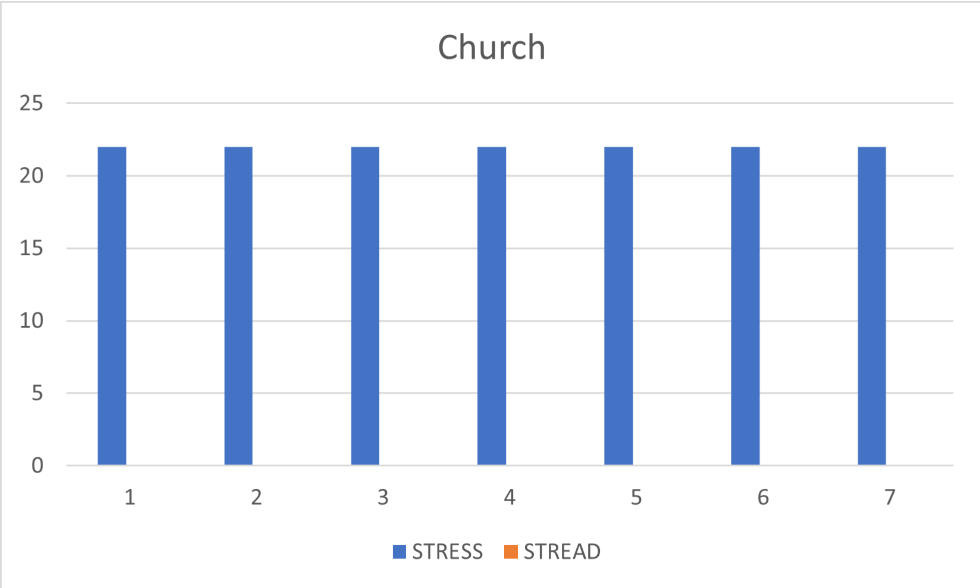


Figure 5.9: Image noise: Church

(a) Church with STREAD **(b)** Church with STRESS

Figure 5.10: Church images with STREAD and STRESS

	Alley	Caps	Church	Flower	Overhead
300	0,000011	0,000011	0,00000048	0,000011	0,00012
350	0,000011	0,000011	0,00000048	0,000011	0,00086
400	0,00012	0,00012	0,00000048	0,00012	0,00012
450	0,0043	0,00012	0,00000048	0,00012	0,017
500	0,0043	0,00086	0,00000048	0,000011	0,000011
550	0,017	0,00012	0,00000048	0,000011	0,000011
600	0,00086	0,00012	0,00000048	0,00012	0,0043

	Red boat	Small alley	Sunrise	Sunset	White flower
300	0,000011	0,00086	0,00000048	0,00000048	0,00000048
350	0,00000048	0,000011	0,000011	0,00000048	0,00000048
400	0,000011	0,0043	0,000011	0,00000048	0,00000048
450	0,00012	0,0043	0,00000048	0,00000048	0,00012
500	0,00086	0,017	0,00000048	0,0043	0,00000048
550	0,00000048	0,0043	0,00086	0,00000048	0,000011
600	0,00012	0,017	0,00086	0,000011	0,00000048

Table 5.5: p-values image noise

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 5.6: p-values each image

displayed in Table 5.6. Here, we observe that the p-values are so small that we can confidently conclude which algorithm is superior for image quality. Finally, a binomial test was carried out on all STREAD and all STRESS, which yielded a p-value $= 8.99 \times 10^{-288}$.

Chapter 6

Discussion

The purpose of this research project was to answer these three research questions:

1. **RQ1:** Can anisotropic diffusion be used for the implementation of envelopes for STRESS?
2. **RQ2:** Does the new implementation create images with better quality than STRESS?
3. **RQ3:** Does the new implementation create images with less image noise than STRESS?

The results of this research project showed that anisotropic diffusion worked really well for the sampling of envelopes and then enhancement of contrast in images, and outperformed the old implementation in terms of objective and subjective measures.

Interpretation of the Results

RQ1 The first research question was about if anisotropic diffusion could be used for sampling instead of stochastic sampling. As we can see in Chapter 3, we were able to implement the technique and use it for contrast enhancement. By tinkering and changing some of the parameters, we were able to get images of great quality and with no image noise. The new implementation was then used to create contrast enhanced images and the images were compared with the ones created with STRESS. The comparison was done with the experiment mentioned in Chapter 4 to give us results for the next research questions.

RQ2 The next question was based on the image quality of the new implementation STREAD, since we wanted to see how the quality compared to STRESS. By doing a paired comparison, we were able to get results based on what image had the best quality. The first result and the plots based on them showed quickly that one of the algorithm were more preferred than the other. STREAD was picked the most for all the images except one, which they were tied. So STREAD was better or equal with STRESS when it came to image quality. By doing binomial tests on the results, with p-values being $p < 0.05$, $p < 0.01$ or $p < 0.001$. All of these thresholds showed added more and more statistical confidence toward STREAD being better than STRESS.

RQ3 The last question was about the image noise in the images, and if STREAD created less noise in the image compared to STRESS. Another paired comparison was done for this question, where almost all the results pointed toward STRESS having the most noise in the image. Which for STREAD showed that it create little to no noise in comparison. The binomial test had even lower p-values than image quality. This also pointed toward STREAD being better than STRESS based on image noise.

Implications

Our results from the research project have implications on both a theoretical level and a practical level. Our results help to expand the current knowledge and understanding of various image enhancement techniques and their impact on image quality from a theoretical perspective. Our results provide empirical evidence that STREAD works better when it comes to image quality and image noise compared to STRESS. The results also shows us how the new implementation works and that it works well with contrast enhancement.

But on the practical level, our results have implications for image processing applications. Examples of such image processing applications are computer vision and medical imaging. In these applications, images with good quality and little noise are really important, since bad quality can ruin the results. The results suggest that using STREAD can improve the quality of contrast enhanced images. It also suggests that STREAD can be used as a preprocessor for other applications, where image quality needs to be precise and not contain any image noise.

Limitations

The research has some limitations that are mainly related to the project's scope and the generalization of the results. First, our research focused on the new sampling method in correlation with contrast enhancement. The result from the project may not be representative for other types of image enhancement techniques or problems. Because of that, our results may not be the same if we use STREAD in other settings like gamut mapping or color correction.

Second, our research used only a psychometric survey to evaluate the results made with STREAD. Metrics based on subjective measures may not capture all aspects of image quality and image noise. The research could have used more objective metrics such as peak signal-to-noise ratio (PSNR) or structural similarity index (SSIM) to support the subjective results. The images used in the experiment can also have an impact on the result, since not all the images picked for the experiment were used because of color correction for one of the algorithms.

Therefore, the results may not be accurate enough to reflect the true performance of STREAD. Even if STREAD is not always significant better than STRESS, we have not found any images where STRESS performs better than STREAD. The two algorithms are either even for some of the images, or STREAD performs better on other images.

Recommendations

Based on these limitations, some recommendations for future research are:

- To test and validate the new implementation on other types of image enhancement techniques or problems mentioned in STRESS [37]. Local color correction of color images, HDR image rendering, spatial color gamut mapping, temporal color correction of movies and color to grayscale conversion.
- To compare STREAD and STRESS with other existing or alternative methods for image enhancement, to figure out the new implementation's strengths and weaknesses.
- To use a larger and more diverse set of images and metrics to evaluate the performance of STREAD.
- With nothing done on STREAD in terms of the calculation speed, make the new algorithm quicker and check how it compares with the speed of STRESS and other alternatives.

Chapter 7

Conclusion

In this research project, we have presented a new implementation based on the STRESS framework that we call STREAD. This new implementation uses anisotropic diffusion for the sampling instead of stochastic sampling used in STRESS. This new implementation was then used for contrast enhancement and compared to STRESS. The project found that STREAD outperforms STRESS in both image quality and image noise in the images generated. STREAD adds to the existing knowledge of image enhancement techniques, and may be used for other enhancement methods down the line, like color correction and spatial gamut mapping. More research is still needed on the implementation, like getting some objective metrics on the results and images. Also, some more testing on the parameters to find the true middle ground that produces the best results. With all this, STREAD can then be applied to the other methods and see how it performs on these methods in comparison with STRESS. Overall, the research project gave good results that will be of good use for future research.

Bibliography

[1] A. S. Parihar and K. Singh, 'A study on retinex based method for image enhancement,' in *2018 2nd International Conference on Inventive Systems and Control (ICISC)*, 2018, pp. 619–624. DOI: 10.1109/ICISC.2018.8398874.

[2] K. Singh and A. S. Parihar, 'A comparative analysis of illumination estimation based image enhancement techniques,' in *2020 International Conference on Emerging Trends in Information Technology and Engineering (ic-ETITE)*, 2020, pp. 1–5. DOI: 10.1109/ic-ETITE47903.2020.195.

[3] S. Jayaraman, S. Esakkirajan and T. Veerakumar, *Digital image processing*. Tata McGraw Hill Education New Delhi, 2009, vol. 7014.

[4] E. H. Land, 'The retinex,' *American Scientist*, vol. 52, no. 2, pp. 247–264, 1964.

[5] C. Párraga, T. Troscianko and D. Tolhurst, 'The human visual system is optimised for processing the spatial information in natural visual images,' *Current Biology*, vol. 10, no. 1, pp. 35–38, 2000, ISSN: 0960-9822. DOI: https://doi.org/10.1016/S0960-9822(99)00262-6. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0960982299002626.

[6] S. Thorpe, D. Fize and C. Marlot, 'Speed of processing in the human visual system,' *nature*, vol. 381, no. 6582, pp. 520–522, 1996.

[7] F. W. Campbell and J. J. Kulikowski, 'Orientational selectivity of the human visual system,' *The Journal of physiology*, vol. 187, no. 2, pp. 437–445, 1966.

[8] G. Simone and I. Farup, 'Spatio-temporal retinex-like envelope with total variation,' Jan. 2012, pp. 176–181.

[9] P. Sagvold, 'Anisotropic diffusion for the implementation of the envelopes for the stress algorithm.'

[10] E. H. Land and J. J. McCann, 'Lightness and retinex theory,' *Josa*, vol. 61, no. 1, pp. 1–11, 1971.

[11] J. J. McCann, J. L. Benton and S. P. McKee, 'Red/white projections and rod/long-wave cone color: An annotated bibliography,' *Journal of Electronic Imaging*, vol. 13, no. 1, pp. 8–14, 2004.

[12] E. H. Land and J. J. McCann, 'Lightness and retinex theory,' *J. Opt. Soc. Am.*, vol. 61, no. 1, pp. 1–11, Jan. 1971. DOI: 10.1364/JOSA.61.000001. [Online]. Available: https://opg.optica.org/abstract.cfm?URI=josa-61-1-1.

[13] E. H. Land, 'The retinex theory of color vision,' *Scientific American*, vol. 237, no. 6, pp. 108–129, 1977, ISSN: 00368733, 19467087. [Online]. Available: http://www.jstor.org/stable/24953876 (visited on 22/05/2023).

[14] J. J. McCann, 'Guest editorial: Special section on retinex at 40,' *Journal of Electronic Imaging*, vol. 13, no. 1, pp. 6–7, 2004.

[15] J. J. McCann and A. Rizzi, *The art and science of HDR imaging*. John Wiley & Sons, 2011.

[16] J. A. Frankle and J. J. McCann, *Method and apparatus for lightness imaging*, US Patent 4,384,336, May 1983.

[17] E. H. Land, 'The retinex theory of color vision,' *Scientific american*, vol. 237, no. 6, pp. 108–129, 1977.

[18] D. Marini and A. Rizzi, 'A computational approach to color adaptation effects,' *Image and Vision Computing*, vol. 18, no. 13, pp. 1005–1014, 2000.

[19] A. Rizzi, D. Marini and L. De Carli, 'Lut and multilevel brownian retinex colour correction,' *Machine Graphics & Vision International Journal*, vol. 11, no. 2/3, pp. 153–168, 2002.

[20] J. J. McCann, 'Capturing a black cat in shade: Past and present of retinex color appearance models,' *Journal of Electronic Imaging*, vol. 13, no. 1, pp. 36–47, 2004.

[21] T. J. Cooper and F. A. Baqai, 'Analysis and extensions of the frankle-mccann retinex algorithm,' *Journal of Electronic Imaging*, vol. 13, no. 1, pp. 85–92, 2004.

[22] G. D. Finlayson, S. D. Hordley and M. S. Drew, 'Removing shadows from images,' in *European conference on computer vision*, Springer, 2002, pp. 823–836.

[23] R. Montagna and G. D. Finlayson, 'Constrained pseudo-brownian motion and its application to image enhancement,' *JOSA A*, vol. 28, no. 8, pp. 1677–1688, 2011.

[24] D. J. Jobson, Z.-u. Rahman and G. A. Woodell, 'Properties and performance of a center/surround retinex,' *IEEE transactions on image processing*, vol. 6, no. 3, pp. 451–462, 1997.

[25] K. Barnard and B. Funt, 'Investigations into multi-scale retinex,' 1998.

[26] F. O. Huck, C. L. Fales, R. E. Davis and R. Alter-Gartenberg, 'Visual communication with retinex coding,' *Applied Optics*, vol. 39, no. 11, pp. 1711–1730, 2000.

[27] J. D. Cowan and P. C. Bressloff, 'Visual cortex and the retinex algorithm,' in *Human Vision and Electronic Imaging VII*, SPIE, vol. 4662, 2002, pp. 278–285.

[28] Z.-u. Rahman, D. J. Jobson and G. A. Woodell, 'Retinex processing for automatic image enhancement,' *Journal of Electronic imaging*, vol. 13, no. 1, pp. 100–110, 2004.

[29] A. Rizzi and C. Bonanomi, 'Milano Retinex family,' *Journal of Electronic Imaging*, vol. 26, no. 3, p. 031 207, 2017. DOI: 10.1117/1.JEI.26.3.031207. [Online]. Available: https://doi.org/10.1117/1.JEI.26.3.031207.

[30] M. Lecca, G. Simone, C. Bonanomi and A. Rizzi, 'Point-based spatial colour sampling in milano-retinex: A survey,' *IET Image Processing*, vol. 12, no. 6, pp. 833–849, 2018.

[31] E. Provenzi, M. Fierro, A. Rizzi, L. De Carli, D. Gadia and D. Marini, 'Random spray retinex: A new retinex implementation to investigate the local properties of the model,' *IEEE Transactions on Image Processing*, vol. 16, no. 1, pp. 162–171, 2007. DOI: 10.1109/TIP.2006.884946.

[32] D. Marini and A. Rizzi, 'A computational approach to color adaptation effects,' *Image and Vision Computing*, vol. 18, no. 13, pp. 1005–1014, 2000, ISSN: 0262-8856. DOI: https://doi.org/10.1016/S0262-8856(00)00037-8. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0262885600000378.

[33] N. Banić and S. Lončarić, 'Light random sprays retinex: Exploiting the noisy illumination estimation,' *IEEE Signal Processing Letters*, vol. 20, no. 12, pp. 1240–1243, 2013. DOI: 10.1109/LSP.2013.2285960.

[34] M. Lecca and A. Rizzi, 'Tuning the locality of filtering with a spatially weighted implementation of random spray retinex,' *J. Opt. Soc. Am. A*, vol. 32, no. 10, pp. 1876–1887, Oct. 2015. DOI: 10.1364/JOSAA.32.001876. [Online]. Available: https://opg.optica.org/josaa/abstract.cfm?URI=josaa-32-10-1876.

[35] G. Gianini, A. Manenti and A. Rizzi, 'Qbrix: A quantile-based approach to retinex,' *J. Opt. Soc. Am. A*, vol. 31, no. 12, pp. 2663–2673, Dec. 2014. DOI: 10.1364/JOSAA.31.002663. [Online]. Available: https://opg.optica.org/josaa/abstract.cfm?URI=josaa-31-12-2663.

[36] G. Gianini, M. Lecca and A. Rizzi, 'A population-based approach to point-sampling spatial color algorithms,' *J. Opt. Soc. Am. A*, vol. 33, no. 12, pp. 2396–2413, Dec. 2016. DOI: 10.1364/JOSAA.33.002396. [Online]. Available: https://opg.optica.org/josaa/abstract.cfm?URI=josaa-33-12-2396.

[37] Ø. Kolås, I. Farup and A. Rizzi, 'Spatio-temporal retinex-inspired envelope with stochastic sampling: A framework for spatial color algorithms,' *Journal of Imaging Science and Technology*, vol. 55, Jul. 2011. DOI: `10.2352/J.ImagingSci.Technol.2011.55.4.040503`.

[38] M. Lecca, C. M. Modena and A. Rizzi, 'Using pixel intensity as a self-regulating threshold for deterministic image sampling in milano retinex: The t-rex algorithm,' *Journal of Electronic Imaging*, vol. 27, no. 1, pp. 011 005–011 005, 2018.

[39] M. Lecca, A. Rizzi and R. P. Serapioni, 'Great: A gradient-based color-sampling scheme for retinex,' *J. Opt. Soc. Am. A*, vol. 34, no. 4, pp. 513–522, Apr. 2017. DOI: `10.1364/JOSAA.34.000513`. [Online]. Available: `https://opg.optica.org/josaa/abstract.cfm?URI=josaa-34-4-513`.

[40] M. Lecca, A. Rizzi and R. P. Serapioni, 'Grass: A gradient-based random sampling scheme for milano retinex,' *IEEE Transactions on Image Processing*, vol. 26, no. 6, pp. 2767–2780, 2017. DOI: `10.1109/TIP.2017.2686652`.

[41] P. Perona, T. Shiota and J. Malik, 'Anisotropic diffusion,' *Geometry-driven diffusion in computer vision*, pp. 73–92, 1994.

[42] P. Perona and J. Malik, 'Scale-space and edge detection using anisotropic diffusion,' *IEEE Transactions on pattern analysis and machine intelligence*, vol. 12, no. 7, pp. 629–639, 1990.

[43] G. Sapiro, *Geometric partial differential equations and image analysis*. Cambridge university press, 2006.

[44] P. Perona and J. Malik, 'Scale-space and edge detection using anisotropic,' 1988.

[45] L. G. Shapiro and G. C. Stockman, *Computer vision*. Pearson, 2001.

[46] D. Mittal, V. Kumar, S. C. Saxena, N. Khandelwal and N. Kalra, 'Enhancement of the ultrasound images by modified anisotropic diffusion method,' *Medical & biological engineering & computing*, vol. 48, no. 12, pp. 1281–1291, 2010.

[47] Y. Yu and S. T. Acton, 'Speckle reducing anisotropic diffusion,' *IEEE Transactions on image processing*, vol. 11, no. 11, pp. 1260–1270, 2002.

[48] K. Z. Abd-Elmoniem, A.-B. Youssef and Y. M. Kadah, 'Real-time speckle reduction and coherence enhancement in ultrasound imaging via nonlinear anisotropic diffusion,' *IEEE Transactions on Biomedical Engineering*, vol. 49, no. 9, pp. 997–1014, 2002.

[49] K. Krissian, 'Flux-based anisotropic diffusion applied to enhancement of 3-d angiogram,' *IEEE transactions on medical imaging*, vol. 21, no. 11, pp. 1440–1442, 2002.

[50] Y. Wang, R. Niu and X. Yu, 'Anisotropic diffusion for hyperspectral imagery enhancement,' *IEEE Sensors Journal*, vol. 10, no. 3, pp. 469–477, 2010.

[51] B. Maiseli, 'Nonlinear anisotropic diffusion methods for image denoising problems: Challenges and future research opportunities,' *Array*, vol. 17, p. 100 265, 2023, ISSN: 2590-0056. DOI: https://doi.org/10.1016/j.array.2022.100265. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2590005622000984.

[52] C. A. Palma, F. A. Cappabianco, J. S. Ide and P. A. Miranda, 'Anisotropic diffusion filtering operation and limitations - magnetic resonance imaging evaluation,' *IFAC Proceedings Volumes*, vol. 47, no. 3, pp. 3887–3892, 2014, 19th IFAC World Congress, ISSN: 1474-6670. DOI: https://doi.org/10.3182/20140824-6-ZA-1003.02347. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S147466701642210X.

[53] D. Tschumperle and R. Deriche, 'Vector-valued image regularization with pdes: A common framework for different applications,' *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 4, pp. 506–517, 2005. DOI: 10.1109/TPAMI.2005.87.

[54] S. Di Zenzo, 'A note on the gradient of a multi-image,' *Computer Vision, Graphics, and Image Processing*, vol. 33, no. 1, pp. 116–125, 1986, ISSN: 0734-189X. DOI: https://doi.org/10.1016/0734-189X(86)90223-9. [Online]. Available: https://www.sciencedirect.com/science/article/pii/0734189X86902239.

[55] I. Farup, 'Individualised halo-free gradient-domain colour image daltonisation,' *Journal of Imaging*, vol. 6, no. 11, 2020, ISSN: 2313-433X. DOI: 10.3390/jimaging6110116. [Online]. Available: https://www.mdpi.com/2313-433X/6/11/116.

[56] I. Farup, 'Variational anisotropic gradient-domain image processing,' *Journal of Imaging*, vol. 7, no. 10, 2021, ISSN: 2313-433X. DOI: 10.3390/jimaging7100196. [Online]. Available: https://www.mdpi.com/2313-433X/7/10/196.

[57] R. Rombach, A. Blattmann, D. Lorenz, P. Esser and B. Ommer, *High-resolution image synthesis with latent diffusion models*, 2022. arXiv: 2112.10752 [cs.CV].

[58] 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*, 2023. arXiv: 2303.09472 [cs.CV].

NTNU
Norwegian University of
Science and Technology