Polarization mapping

Fernando del Molino, Adolfo Muñoz

Universidad de Zaragoza

ARTICLE INFO

ABSTRACT

Photographers use the hardware of the camera (aperture, exposure time...), lens and filters as tools for artistic expressivity. This expressivity has often been enhanced by software, such as high dynamic range images have been edited in post-process with software tone mappers. In this paper, we propose a similar approach with polarization filters: we design a capture process that enables us to acquire a Stokes image (that encodes all the possible light polarization states) with a single camera, and we then offer a set of software tools that can apply any common polarization filter as a software post-process, delaying the choice of the adequate filter and enabling filters that can be mathematically modeled but are not available as hardware. Then, we devise and provide new algorithms that automatically select the optimal filter for specific goals, such as maximizing (or minimizing) brightness, contrast or saturation. We later show how such optimization filters can not only be applied to the whole image, but can also be at per-pixel level, obtaining new interesting effects. Such optimization can work at real time rates, fact that is illustrated with a brush based user editing interactive tool. The different types of filters are tested in a wide range of results.

1. Introduction

Computational photography is a new and multidisciplinary research field that involves a diverse range of disciplines such as computer vision, image processing, computer graphics or applied optics. It refers broadly to computational imaging techniques that enhance or extend the capabilities of digital photography. Recent advances in this field have given rise to new digital imaging tools. By acquiring more complete scene data, post-capture software image processing techniques can be applied to obtain and extend certain effects that were traditionally restricted to camera hardware.

One example of such extended scene data is a High Dynamic Range (HDR) image. Often composed of a set of photos of the same scene at varied exposure times, they present a huge resolution in the luminance dimension (HDR stores a floating-point per channel instead of the standard RGB 8-bit per channel). Having an HDR as input, post-processing algorithms can, for instance, simulate the effect of exposure time after the photographs has already been taken. This often leads to results that emulate the behavior of photographic cameras, or even to effects that are plainly impossible to obtain with such hardware, through advanced non-local tone mapping operators. Furthermore, HDR images open a broad set of new editing tools that were not possible before [1].

Another example of such extended image representations are light fields [2], which are four-dimensional captures of the scene, where each pixel is demultiplexed in the (two) angular dimensions. This representation is often obtained using specialized cameras such as plenoptic cameras, and enables post-processing effects such as shifting the point of view of the

e-mail: ferdelmo1996@gmail.com (Fernando del Molino),
adolfo@unizar.es (Adolfo Muñoz)

scene, switching the focal point [3] and generating depth of field effects, again emulating photographic hardware by software.

In this work, we propose an approach similar to HDR images and light fields by extending the scene representation to Stokes images, in which each pixel (and each channel) encodes all possible polarization states of light (or even unpolarized light). Light is an electromagnetic wave that oscillates perpendicularly to its propagation direction. Generally speaking, a beam of light is unpolarized (it carries a random mixture of all potential polarizations), but there are several circumstances in nature (atmospheric scattering, specular reflections, refractions or polarization filters) in which light gets (at least partially) coherent (polarized). While the human eye is oblivious to light polarization, polarization filters are often used in photography for specific effects (see Figure 1).

Fig. 1: Real photographs obtained with different polarization filters. Left image is an image without any polarizer and right image with a linear polarizer, where the sky tone has notorious changes, modifying the saturation. Another common use of polarizers is to modify the reflections on different specular surfaces.

Our work enables a photographer to postpone the decision of the optimal polarizing filter to post-processing. If the effect of such filter is applied in photo editing software, the user can always remove it or change it to taste, exaggerating or even creating effects that are not possible with current hardware. We provide a low-cost approach for capturing such Stokes image using consumer hardware and a set of software operators for emulating polarizing filters from such Stokes images. Given a Stokes image, we provide the following post-processing operators:

- Standard linear and circular virtual polarization filters, which emulate the use of hardware polarization filters on a camera.

- Elliptical and combined polarization filters, which are not produced nor commercialized, but can be applied in software.

- Automatic filters, which maximize (or minimize) specific features of the image (contrast, specular glares, saturation) by applying the optimal polarization filter on each case.

- Automatic local filters, which maximize (or minimize) those features locally applying a different polarization to each pixel, which are hardly obtainable using consumer hardware (it would require specific hardware [4]).

All filters, except local saturation and contrast, can work in real time, enabling interactive usage. We show this with an interactive brush based tool that can apply all filters mentioned above.

2. Related work

Polarimetric imaging. Polarimetric information has been used to study different and diverse fields. In computer vision, for instance, polarimetric images have been applied to eliminate the haze of an image [5] or specular highlights [6, 7] using polarization. With a different goal (general photographic editing), our approach is able to eliminate haze or highlights as a consequence of luminance minimization, while providing several other editing operators.

Both in computer vision and computer graphics, polarimetric imaging has been used for a while to study material properties, such as classifying material as dieletic or metals [8] or estimating reflectance properties [9, 10]. Related to this, the work by Toisul and Dhillon [11] uses a polarization based method to capture the appearance of printed holographic surfaces.

Polarimetric imaging has also been applied for capturing geometry, such as the general shape of an object [12, 13] or the normals of a surface [14, 10]. This also presents some similarities with our approach, as our local luminance maximization finds the angles at which per-pixel luminance is maximized, which are related to surface normals. However, our approach is focused on image processing and not on feature estimation.

Osante's Master Thesis [15] introduces the key idea of applying polarizing filters by software, but, as opposed to our approach, only global filters are applied and only brute force global optimization is considered.

Polarimetric imaging is also used on other diverse fields. It has been used in cosmology to measure the polarized spectrum of massive black holes [16]. Biomedical engineering is another field where polarimetric imaging is used. Polarized backscattering light can carry information about a tumor, useful information for cancer diagnosis [17].

Stokes parameters acquisition. The state of polarization can be measured with a polarimeter. Since the introduction of the Stokes parameters in 1852, different methods to capture the polarimetric information of an image have been proposed. North and Duggin [18] present a practical method to capture partial Stokes vector (without the circular component) of the polarized sky-dome using a four-lens stereoscopic camera. In addition to all this concrete purpose methods, general Stokes cameras have been recently proposed. Tu et al. [4] introduce a method to recover precisely full RGB Stokes parameters using micro-polarizers arrays, requiring complex calibration. Vedel et al. [19] also propose a method that can recover full Stokes parameters capturing four different states of polarization. Despite the fact there are several methods to capture full Stokes parameters, there is still no accessible commercial camera to capture RGB Stokes parameters. How to acquire the Stokes parameters of a scene was already introduced in previously mentioned work, and our capture method is based on it. Our capture method avoids the use of complex and personalized methods to ensure accessibility and simplicity, as it only uses consumer hardware, which makes it cheaper, and can be easily used by any amateur photographer.

3. Polarization

Electromagnetic waves, such as light, are transverse waves: they oscillate in a perpendicular plane to the propagation direction, known as the polarization plane. In natural light, these oscillations happen simultaneously on all directions with equal probability. However, there are some interactions, both natural and artificially induced, that may force some coherency into the oscillation of the wave. Such coherency is called polarization. The interactions that polarize waves are known as polarizing events, and include natural sources such as dielectrics (like water or glass), and artificial polarization filters, that force specific polarization states on electromagnetic waves that traverse them (see Figure 2) [20, 21].
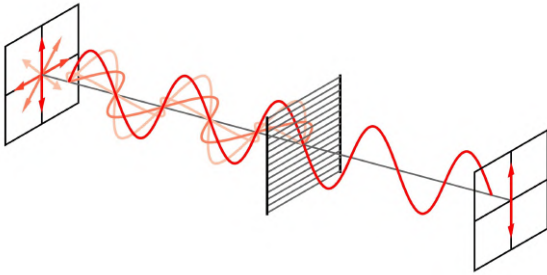
Fig. 2: Propagation of natural light across a linear polarizer. The vertical components are transmitted, while the horizontal components are absorbed and reflected.

The polarization state of a wave can be described as the temporal evolution of the electric field in a certain perpendicular plane of propagation direction z. The electric field is described by a vector $\vec{E}$, that can be represented as two perpendicular components, $E_x(z,t)$ and $E_y(z,t)$ (t represents time). They are expressed as:

$$
\begin{aligned}
E_x(z,t) &= E_{0x}\cos(Kz - \omega t + \delta_x) \\
E_y(z,t) &= E_{0y}\cos(Kz - \omega t + \delta_y)
\end{aligned}
\tag{1}
$$

where E_{0x} and E_{0y} are the amplitudes on both axis, ω is the angular frequency, K is the wave number, and δ_x, δ_y are absolute phases on the x and y directions, respectively.

Equation (1) is not particularly revealing because the field components can not be directly observed nor understood. A useful visual representation of the polarization behavior can be obtained by eliminating the time-space propagator $(Kz - \omega t)$ and by therefore projecting the shape of the electromagnetic wave into the polarization plane.

From this projection, fully polarized light can be classified by the shape of this ellipse, that depends on the values of $\delta = \delta_x - \delta_y$, E_{ox} and E_{oy}. Three different types of full polarization are classified by this method (see Figure 3):

- **Linear polarization**: the temporal evolution of the electric field shapes a line, when $\delta = n \cdot \pi$ where $n = 0, 1, 2, ..$

- **Circular polarization**: the temporal evolution of the electric field shapes a circle, when $\delta = \frac{\pi}{2} + n \cdot \pi$ where $n = 0, 1, 2, ..$ and $E_{ox} = E_{oy}$

- **Elliptical polarization**: the temporal evolution of the electric field shapes a ellipse, for the rest of the cases.
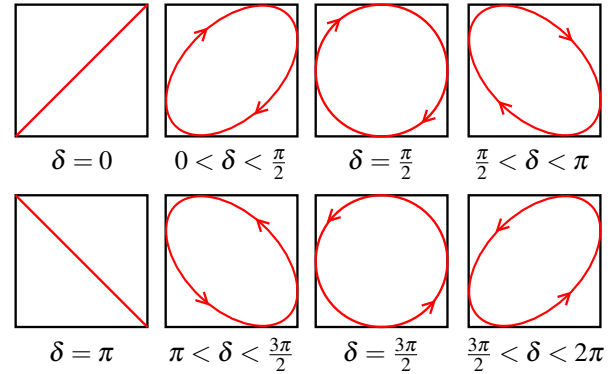
Fig. 3: Depending on the δ value, the projected ellipse (polarization ellipse) has a different shape ($E_{ox} = E_{oy}$)

3.1. Polarization representation

Still, the type of polarization (obtained from the projection of the wave to the polarization plane) can not be observed nor measured [22]. Furthermore, electromagnetic waves may not just be either natural or fully polarized: they can be partially polarized as well. To fully determine the polarization observable parameters they must be expressed in the intensity domain [23]. One of such representations is the Stokes vector.

The Stokes vector $\mathbf{S}$ is defined by four parameters (I, Q, U, V) that are described in terms of intensities (amplitudes squared) and therefore can be measured. Each parameter has a specific meaning:

- I represents the total intensity of the beam.

- U describes the predominance of the linearly horizontally polarized (I_0) light over linearly vertically polarized (I_{90}) light.

- V describes the predominance of linearly diagonally polarized (I_{45}) light over the perpendicular diagonally polarized (I_{135}) light.

- Q describes the preponderance of right-circularly polarized (I_{RCP}) light over left-circularly polarized (I_{LCP}) light.

The relation between such Stokes parameters and the electromagnetic wave representation shown in Equation (1) can be expressed, for a fully polarized wave, as:

$$
\mathbf{S} = \begin{pmatrix} I \\ Q \\ U \\ V \end{pmatrix} = \begin{pmatrix} E_{0x}^2 + E_{0y}^2 \\ E_{0x}^2 - E_{0y}^2 \\ 2 \cdot E_{0x} \cdot E_{0y} \cdot \cos\delta \\ 2 \cdot E_{0x} \cdot E_{0y} \cdot \sin\delta \end{pmatrix}.
\tag{2}
$$

However, Stokes parameters not only describe completely polarized light but also unpolarized and partially polarized light as well, where

$$
I^2 \geq Q^2 + U^2 + V^2 \geq 0
\tag{3}
$$

Alternatively to the representation shown on Equation (2), a Stokes vector can be generated from six degenerate polarization states:

$$\mathbf{S} = \begin{pmatrix} I \\ Q \\ U \\ V \end{pmatrix} = \begin{pmatrix} I_0 + I_{90} \\ I_0 - I_{90} \\ I_{45} - I_{135} \\ I_{RCP} - I_{LCP} \end{pmatrix}, \qquad (4)$$

where I_α is the intensity of the light beam transmitted across a linear polarizer oriented at an angle α with respect to the x-axis and $I_{RCP,LCP}$ is the right- or left-handed circularly polarized component of the intensity. Note that $I = I_0 + I_{90} = I_{45} + I_{135} = I_{RCP} + I_{LCP}$. This representation is very useful for capture purposes (see Section 4). Degenerate polarization states are singular, as they are fully polarized ($I^2 = Q^2 + U^2 + V^2$) with all Q, U and V, except when one has a zero value. Such degenerate polarization states are illustrated in Figure 4.
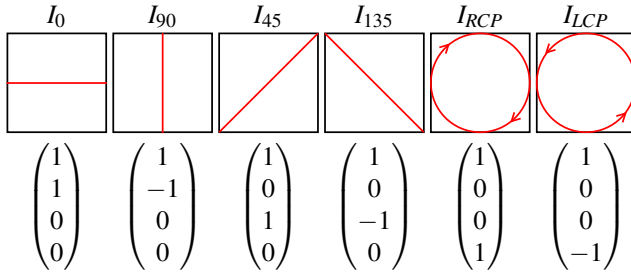

Fig. 4: Stokes parameters for degenerate polarization states. Each state is fully polarized, with only one of the parameters Q, U or V has a value different from zero.

3.2. Polarizing elements and Mueller calculus

Polarization state can be changed by polarizing elements, either natural (dielectric or reflective surfaces, the sky) or artificial (polarizing filters). Mathematically, any polarizing element can be represented as a 4×4 matrix $\mathbf{M}$, called Mueller matrix. Any polarizing interaction can be represented as

$$\mathbf{S'} = \begin{pmatrix} I' \\ Q' \\ U' \\ V' \end{pmatrix} = \mathbf{M} \cdot \mathbf{S} = \mathbf{M} \begin{pmatrix} I \\ Q \\ U \\ V \end{pmatrix} \qquad (5)$$

where $\mathbf{S}$ represents the Stokes vector of the incident beam, $\mathbf{M}$ the Mueller matrix of the polarizing element and $\mathbf{S'}$ the Stokes vector of the outgoing beam.

One of the advantages of such notation is that the combination of multiple (n) polarizing elements with specific Mueller matrices $\mathbf{M_i}, i \in [1..n]$, can be expressed as a matrix product:

$$\mathbf{S'} = \left(\prod_{i=1}^{n} \mathbf{M_i} \right) \mathbf{S} = \mathbf{M_1} \mathbf{M_2} ... \mathbf{M_n} \mathbf{S}. \qquad (6)$$

There are three basic artificial polarizing elements, that combined can generate more sophisticated filters. These can modify all the configuration parameters of the polarization ellipse: the orthogonal amplitudes and the phase difference. The amplitude can be changed by using a polarizing element known as a polarizer. Similarly, the phase of an optical beam can be changed by a wave plate (also called a retarder or phase shifter). Finally, the polarization ellipse can be changed by rotation using a component called a rotator. Using these three polarizing elements, any elliptical polarization state can be obtained. The corresponding Mueller matrices of these 3 elements are:

- **Linear polarizer.** It is an anisotropic attenuator that attenuates the orthogonal components of a light beam unequally (with p_x and p_y attenuation factors):

$$\mathbf{M_{LP}}(p_x, p_y) = \frac{1}{2} \begin{pmatrix} p_x^2 + p_y^2 & p_x^2 - p_y^2 & 0 & 0 \\ p_x^2 - p_y^2 & p_x^2 + p_y^2 & 0 & 0 \\ 0 & 0 & 2p_x p_y & 0 \\ 0 & 0 & 0 & 2p_x p_y \end{pmatrix} \qquad (7)$$

- **Wave plate.** Introduces a phase shift of δ between the orthogonal components of the incident light beam:

$$\mathbf{M_{WP}}(\delta) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & \cos\delta & \sin\delta \\ 0 & 0 & -\sin\delta & \cos\delta \end{pmatrix} \qquad (8)$$

- **Rotator.** Rotates the orthogonal field components (E_x, E_y) by an angle α:

$$\mathbf{M_{ROT}}(\alpha) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos 2\alpha & \sin 2\alpha & 0 \\ 0 & -\sin 2\alpha & \cos 2\alpha & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \qquad (9)$$

4. Stokes images capture

For the purpose of this work, we need to capture and store the complete polarization state of a scene. To achieve this goal, we use an image format that has 12 channels per pixel, which allows us to store the complete polarization state of every pixel in the image, since we save the 3 components RGB for the 4 Stokes parameters. To capture this new image format, we present a cheap method that only needs two polarizer filters, one linear and one circular, a camera and a tripod.

The key idea is to start from Equation (4) but instead of using a single intensity value, we capture a filtered image according to the specifications. In this case, six different images would be needed in order to capture the complete state of polarization of a certain scene (I_0, I_{45}, I_{90}, I_{135}, I_{RCP} and I_{LCP}), that require 3 different filters: one linear polarizer and two different circular polarizer (RCP and LCP).

By reformulating Equation (4), it is possible to express the parameters Q, U and V as a function of I (the image without any filter), as shown in Equation (10).

$$\mathbf{S} = \begin{pmatrix} I \\ Q \\ U \\ V \end{pmatrix} = \begin{pmatrix} I \\ 2I_0 - I \\ 2I_{45} - I \\ 2I_{RCP} - I \end{pmatrix} \qquad (10)$$

This way, we can reduce the number of captures to four. The limitations of this approach, however, are that the polarizing filters and the optics of the system are not ideal, and they attenuate light in an uncontrollable way (often too much compared to the ideal). Therefore, in the end we find a much more stable way that does not require such calibration, because it does not require an image without filter:

$$\mathbf{S} = \begin{pmatrix} I \\ Q \\ U \\ V \end{pmatrix} = \begin{pmatrix} (I_0 + I_{90} + I_{45} + I_{135})/2 \\ I_0 - I_{90} \\ I_{45} - I_{135} \\ 2I_{RCP} - I \end{pmatrix} \quad (11)$$

The approach in Equation (11) requires five captures, four of them with a linear polarizer (rotated at four different angles) and the other requires a circular polarizer. In photography, the circular polarizers, known as CPLs, are inverted with respect to the theoreticals (the side to attach the CPL to the camera lens should be the other to filter circularly polarized light). A circular polarizer is constructed from a linear polarizer and a quarter wave plate:

$$\mathbf{M_{RCP}} = \mathbf{M_{LP}}(45°)\mathbf{M_{QWP}}$$
$$\mathbf{M_{LCP}} = \mathbf{M_{LP}}(-45°)\mathbf{M_{QWP}} \quad (12)$$

where $\mathbf{M_{RCP}}$ and $\mathbf{M_{LCP}}$ are the Mueller matrix for right and left polarizers. $\mathbf{M_{LP}}(\alpha)$ is a linear polarizer tilted α degrees with respect the horizontal line. $\mathbf{M_{QWP}}$ is the Mueller matrix of a quarter wave plate, where $\delta = \pi/2$ (see Equations (7) and (8)).

Unlike linear polarizers, circular polarizers do not have the same effect from the two sides (matrix multiplication is not commutative). This means that standard CPLs do not filter circularly polarized light. We flip the crystal of the CPL to obtain a pure circular polarizer. As RCP and LCP are indistinguishable, we assume our filter behaves like a RCP to calculate V. If it does not, the only difference is the sign in V, which does not limit the filters that can be applied (see Equations (4) and (11)).

An example of a scene captured by this method can be seen in the Figure 5. The hardware used to take the images is: a Canon EOS 20D camera, a TAMRON SP AF 17-50mm F/2.8, a HOYA Linear Polarizer 67mm and a HOYA Circular Polarizer 67mm.

Last, as polarizers reduce by half the brightness of the image, we adjust the exposition so the luminance radiance is comparable to the unfiltered scene. This is also done by photographers when using polarization filters.

Note that our capture approach, and as a consequence the later photographic edits, are independent of the emitted light's polarization from the light sources: it can be unpolarized or very strongly polarized, and the resulting Stokes image will contain the interaction of such emitted light with the objects in the scene, no matter its polarization state.

5. Polarization mapping

We introduce the concept of polarization mapping, analogous to tone mapping for HDR images, but applying polarizing filters by software as a post-process to a captured Stokes image.
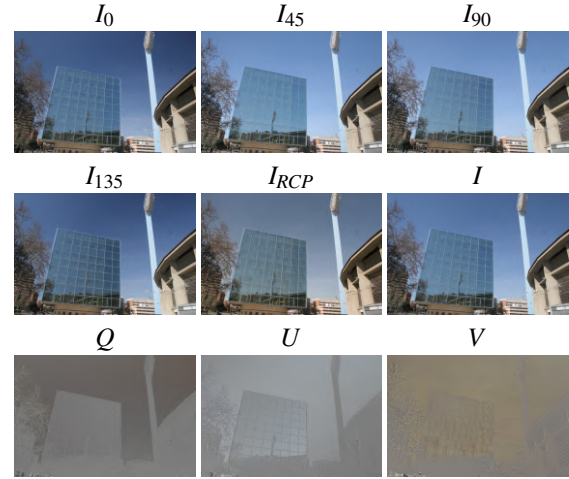

Fig. 5: Visualization of the Stokes parameters of a scene and the original images that formed it. I_0, I_{45}, I_{90}, I_{135} and I_{RCP} are the images taken with the camera. I, Q, U and V are the parameters of the generated Stokes image. As Q, U and V can have values between $[-1, 1]$, they can't be visualized, so in the figure these parameters are normalized as: $Q_{nor} = \frac{Q+1}{2}$

5.1. Software polarizing filters

Once we have captured a Stokes image, we have all the potential polarization states encoded into a single 12-channel image format. Furthermore, for each pixel and each RGB color channel, the I component of the Stokes vector represents an image that can be visualized (see Section 3.1). Additionally, Mueller calculus enables the application of any polarizing element to a Stokes vector by means of a 4×4 Mueller matrix $\mathbf{M}$. For editing in post-process, the only output needed is the resulting I' value per pixel $\mathbf{p}$ for the specific Mueller matrix that represents the polarization element, hence we only need to multiply the Stokes vector at each pixel $\mathbf{p}$ and RGB channel c by the first row of the matrix:

$$I'(c, \mathbf{p}) = m_{00}I(c, \mathbf{p}) + m_{01}Q(c, \mathbf{p}) + m_{02}U(c, \mathbf{p}) + m_{03}V(c, \mathbf{p}) \quad (13)$$

Where $I'(c, \mathbf{p})$ is the outgoing value per channel c and per pixel $\mathbf{p}$, m_{0y} are the coefficients of the first row of the Mueller matrix that represents the polarizing element and $I(c, \mathbf{p})$, $Q(c, \mathbf{p})$, $U(c, \mathbf{p})$ and $V(c, \mathbf{p})$ are the input Stokes parameters in the Stokes image format for pixel $\mathbf{p}$ and RGB channel c. For brevity, the dependency on the RGB channel c is omitted in the rest of the text, but still all calculations are performed per channel too.

By modeling real hardware polarizing filters by their specific Mueller matrix, incorporating them into Equation (13) and applying them for each pixel $\mathbf{p}$ of the image, we can obtain a new image that is the result of applying the polarizing element. This is done by software, as a post-process, and not by hardware.

For instance, a **linear polarizer** with matrix $\mathbf{M_{LP}}(\alpha)$ (where α is the rotation angle of the filter with respect to the horizontal axis) is modeled as:

$$\mathbf{M_{LP}}(\alpha) = \mathbf{M_{LP}}(1, 0)\mathbf{M_{ROT}}(\alpha) \quad (14)$$

$\mathbf{M_{LP}}$ and $\mathbf{M_{ROT}}$ are the Mueller matrix of a linear polarizer and a rotator respectively (defined in Equations (7) and (9)).

Fig. 6: Different images refiltered with linear and elliptical filters. First row shows different images refiltered with a linear polarizer at different angles (from left to right, the angle of the linear polarizer increases from $0°$ to $150°$ with steps of $30°$). Different orientations of the linear polarizer control which reflections of the car are visible. Second row shows different images refiltered with an elliptical polarizer at different rotations (from left to right, α increases from $0°$ to $150°$ with steps of $30°$) while $\delta = \pi/2$ constant.

By plugging the corresponding coefficients of $\mathbf{M_{LP}}$ (the Mueller matrix of a linear polarizer) into Equation (13) we obtain:

$$I'(\mathbf{p},\alpha) = \frac{1}{2}(I(\mathbf{p}) + Q(\mathbf{p})\cos(2\alpha) + U(\mathbf{p})\sin(2\alpha)) \quad (15)$$

that can, given a filter angle α, obtain the result of applying such hardware filter for each pixel $\mathbf{p}$ of the image. This is illustrated in Figure 6.

The same methodology can be applied for modeling a commercial **circular polarizer**, which is often build as Equation (12). Specifically, a general elliptical polarizer constructed from a linear polarizer rotated at α degrees with a δ wave-plate can be calculated as:

$$I'(\mathbf{p},\alpha,\delta) = \frac{1}{2}(I(\mathbf{p}) + Q(\mathbf{p})\cos(2\alpha) + U(\mathbf{p})\sin(2\alpha)\cos(\delta)$$
$$+ V(\mathbf{p})\sin(2\alpha)\sin(\delta)) \quad (16)$$

Circular polarizers (right and left) are obtained with $\delta = \frac{\pi}{2}$ and $\alpha = \pm\frac{\pi}{4}$. Still, the linear filter can be rotated (α angle), generating filtered images such as the ones shown in Figure 6.

Additionally, arbitrary Mueller parameters $m_{0y}, y \in [0,3]$ generate custom filters that are hardly reproducible in hardware. While exploring these would provide new results, as such parameterization is the most versatile, an arbitrary choice of parameters provides no new insight, so we consider such analysis out of the scope of this paper.

5.2. Global optimization filters

In photography, polarizing filters are used for various situations. Its most common uses are to modify the specular reflections on specific surfaces of the scene, to explore different color values for polarizing media such as the sky, or to maximize contrast or saturation in the image. The photographer rotates the chosen polarizing filter (linear polarizer) trying different angles until the desired goal is achieved. Taking advantage of a Stokes image, we do the same by formulating such goal as an **optimization problem**.

Specifically, we define a methodology based on rotating a linear polarizer according to the angle α. We define a target function $f(\alpha)$ to optimize (maximize or minimize), by analytically calculating the first ($f'(\alpha)$) and second ($f''(\alpha)$) derivative with respect to the angle α. By finding the roots of the derivative ($f'(\alpha) = 0$) we get the critical points, and by analyzing the second derivative for each critical point we obtain whether it is a maximum ($f''(\alpha) < 0$) or minimum ($f''(\alpha) > 0$).

Luminance. Luminance is a powerful scene descriptor in terms of polarization. By minimizing luminance we actually minimize the highlights on different reflective surfaces. Our luminance target function is:

$$L(I,\mathbf{p}) = 0.2126 I(R,\mathbf{p}) + 0.7152 I(G,\mathbf{p}) + 0.0722 I(B,\mathbf{p}) \quad (17)$$
$$L(I) = \sum_{\mathbf{p} \in \mathbf{P}} L(I,\mathbf{p}), \quad (18)$$

where $\mathbf{P}$ represents all the pixels in the image.

By combining Equations (18) and (15), we define $L(I,\alpha)$, the luminance in terms of a linear polarizer, and finding the roots of the corresponding derivative with respect to the angle α we get two possible values:

$$\alpha = \frac{1}{2}\arctan\left(\frac{\sum_{\mathbf{p}\in\mathbf{P}} L(U,\mathbf{p})}{\sum_{\mathbf{p}\in\mathbf{P}} L(Q,\mathbf{p})}\right)$$
$$\alpha = \frac{1}{2}\arctan\left(\frac{\sum_{\mathbf{p}\in\mathbf{P}} L(U,\mathbf{p})}{\sum_{\mathbf{p}\in\mathbf{P}} L(Q,\mathbf{p})}\right) + \frac{\pi}{2} \quad (19)$$

By analyzing the second derivative we find out which one is the maximum and which one the minimum, and apply the corresponding linear filter, as shown in Section 5.1. Results of this approach are shown in Figure 7. This is equivalent to what the photographer would get by rotating the filter accordingly.

Contrast. Contrast measures the difference in luminance (or color) that makes objects distinguishable. Unlike luminance, the contrast C at each pixel depends on other pixels that surround it. For global contrast optimization, we use the Root Mean Square contrast formula for luminance, which is related to the standard deviation of the luminance on the image:

$$\mu(I,X) = \frac{1}{|X|} \sum_{\mathbf{p}\in X} L(I,\mathbf{p})$$
$$C(I) = \sqrt{\frac{1}{|\mathbf{P}|} \sum_{\mathbf{p}\in\mathbf{P}} (L(I,\mathbf{p}) - \mu(I,\mathbf{P}))^2} \quad (20)$$

where $\mu(I, X)$ is the mean of the channel I in the set of pixels X

For finding the local minima and maxima, we apply the same methodology: we obtain a target function from Equations (20) and (15), and the roots of its derivative are the potential minima and maxima. It becomes a fourth degree polynomial equation:

$$\cos^4(2\alpha) + a\cos^3(2\alpha) + b\cos^2(2\alpha) + c\cos(2\alpha) + d = 0 \quad (21)$$

The values of the coefficients and the full derivation can be found at the supplementary material.

With a change of variable $x = \cos 2\alpha$, we find the roots using the Descartes-Euler method [24], and change it back to α. A fourth degree polynomial can have: 4 real roots, 2 real roots and 2 imaginary or 0 real roots and 4 imaginary. Given the change of variable each real root yields two different α values to check. We compare all of them, choosing the α that gives the maximum (or minimum) value. If no real roots exist, we select the best angle from a predetermined set, $\{10, 20, 30, .., 170\}$. Our experiments show that this happens on less than 0.1% of the cases. The results of contrast optimizations are shown in Figure 7.

Saturation. Saturation measures the intensity of a specific hue in a color. A saturated color is perceived as a bright and intense color, while the grayscale is the least saturated. In an RGB color system, saturation can be described as the standard deviation of the color space

$$\mu_c(I, \mathbf{p}) = \frac{I(R, \mathbf{p}) + I(G, \mathbf{p}) + I(B, \mathbf{p})}{3}$$
$$S(I, \mathbf{p}) = \sqrt{\frac{(I(R,\mathbf{p}) - \mu_c(I,\mathbf{p}))^2 + (I(G,\mathbf{p}) - \mu_c(I,\mathbf{p}))^2 + (I(B,\mathbf{p}) - \mu_c(I,\mathbf{p}))^2}{3}}$$
$$(22)$$

$$S(I) = \sum_{\mathbf{p} \in \mathbf{P}} S(I, \mathbf{p}) \quad (23)$$

where $\mu_c(I, \mathbf{p})$ is the mean of the three channels at pixel $\mathbf{p}$ of image I, $S(I, \mathbf{p})$ is the saturation value for pixel $\mathbf{p}$ of image I and $S(I)$ is the total saturation of the image. Note that this definition is actually not the best saturation definition for the RGB color space, and that actually other more robust saturation metrics involve the choice of primary colors, the white point and / or calculating minima or maxima. Such functions are not differentiable, but Equation (23) can be differentiated and hence is suitable for our methodology.

Once more, combining the total saturation $S(I)$ with the linear polarization filter of Equation (15) we obtain the target function whose derivative presents maxima and minima for the filter rotation α as its roots. As it happens with the contrast optimization, to find the best α values we have to find the roots of a fourth degree polynomial, and we apply the same procedure as for contrast. The full derivation can be found at the supplementary material. The results of maximizing and minimizing saturation are shown in Figure 7.

Note that, while our methodology is based on rotating linear polarizers, the same methodology could be applied for optimizing the rotation of other filters (circular polarizers, for instance) or even arbitrarily parameterized filters, which are out of the scope of this work. More results of these three filters can be seen in our supplemental material.

5.3. Local optimization filters

Optimization filters enable to find the angles at which certain target functions (luminance, contrast or saturation) are maximized or minimized, emulating what the photographer does when rotating the filter searching for the desired artistic effect. In this case, however, we have all the polarization information at per-pixel level, so it is mathematically possible to apply a different Mueller matrix to each pixel. In the same spirit that there exists global and local tone-mapping operators, we also introduce local polarization mapping, in which each pixel is treated independently. Specifically, we find a way to optimize the same specific target functions (for luminance and saturation, for contrast we define a new metric) at per-pixel level, in which each pixel is filtered with a linear polarization filter at a different angle. This provides a new degree of flexibility and enables a new set of interesting effects.

To visualize the filters that are selected for each pixel, we use the HSV color space, where the Hue will be the angle selected for each pixel, while values for Saturation and Value are constant.

Luminance. For finding the optimal angle that maximizes (or minimizes) luminance at per-pixel level, we start from the local pixel definition of luminance, shown in Equation (17). We follow the same optimization methodology of finding the roots of the derivative with the linear filter described in Equation (15), with the difference that in this case it is applied to obtain the optimal angle at each pixel. The resulting optimal values are:

$$\alpha(\mathbf{p}) = \frac{1}{2}\arctan\left(\frac{L(U, \mathbf{p})}{L(Q, \mathbf{p})}\right), \quad \alpha(\mathbf{p}) = \frac{1}{2}\arctan\left(\frac{L(U, \mathbf{p})}{L(Q, \mathbf{p})}\right) + \frac{\pi}{2}$$
$$(24)$$

As the luminance reflected by a surface in a certain point is highly related to its normal, the change of the angle in neighbor pixels will be as smooth as the change of the normal on the surface. Local luminance filters are better than their global counterparts at eliminating specular reflections, because in some scenes, such as in Figure 8, different reflecting surfaces (dielectrics or similar) are oriented towards different directions. As polarization after the interaction depends on surface orientation, the maximum and minimum angles differ on both surfaces. As such, a global optimization, as it only selects a single angle, is unable to deal simultaneously with both of them. However, the local luminance optimization handles them flawlessly and independently.

Contrast. Instead of the global metric, for each pixel $\mathbf{p}$ we define a Gaussian window of pixels around it ($\mathbf{W_p}$), leading to the following contrast metric

$$C(I, \mathbf{p}) = \sqrt{\frac{1}{|\mathbf{W_p}|}\sum_{\mathbf{q} \in \mathbf{W_p}} w(\mathbf{p}, \mathbf{q})(L(I, \mathbf{q}) - \mu(I, \mathbf{W_p}))^2} \quad (25)$$

where $w(\mathbf{p}, \mathbf{q}) = e^{\frac{|\mathbf{q} - \mathbf{p}|^2}{2\sigma^2}}$ and σ is the standard deviation of the Gaussian distribution.

Such window depends on two parameters, the radius, which delimits the pixels that belong to the window, and σ, to adjust the weight of each pixel depending on its distance to the center. We use $\sigma \approx \frac{r}{3}$.
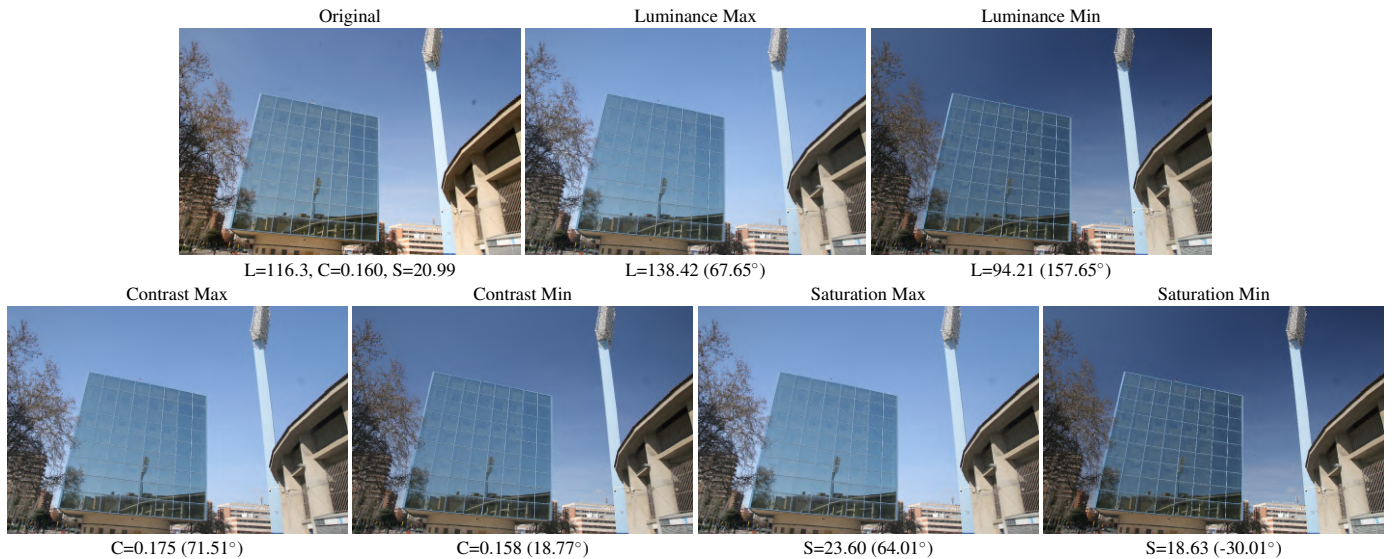
Fig. 7: Illustration of our global optimizing filters. Below each image, the optimized value and the angle of the linear polarizer selected are shown. The luminance filter achieve to minimize or maximize the highlights in the windows of the building. In the contrast filter, the main change can be seen in the sky. Different sky tones are achieved by the saturation filter. These images are 1080x720px, and the refilter time is 180ms for the luminance and 500ms for contrast and saturation.
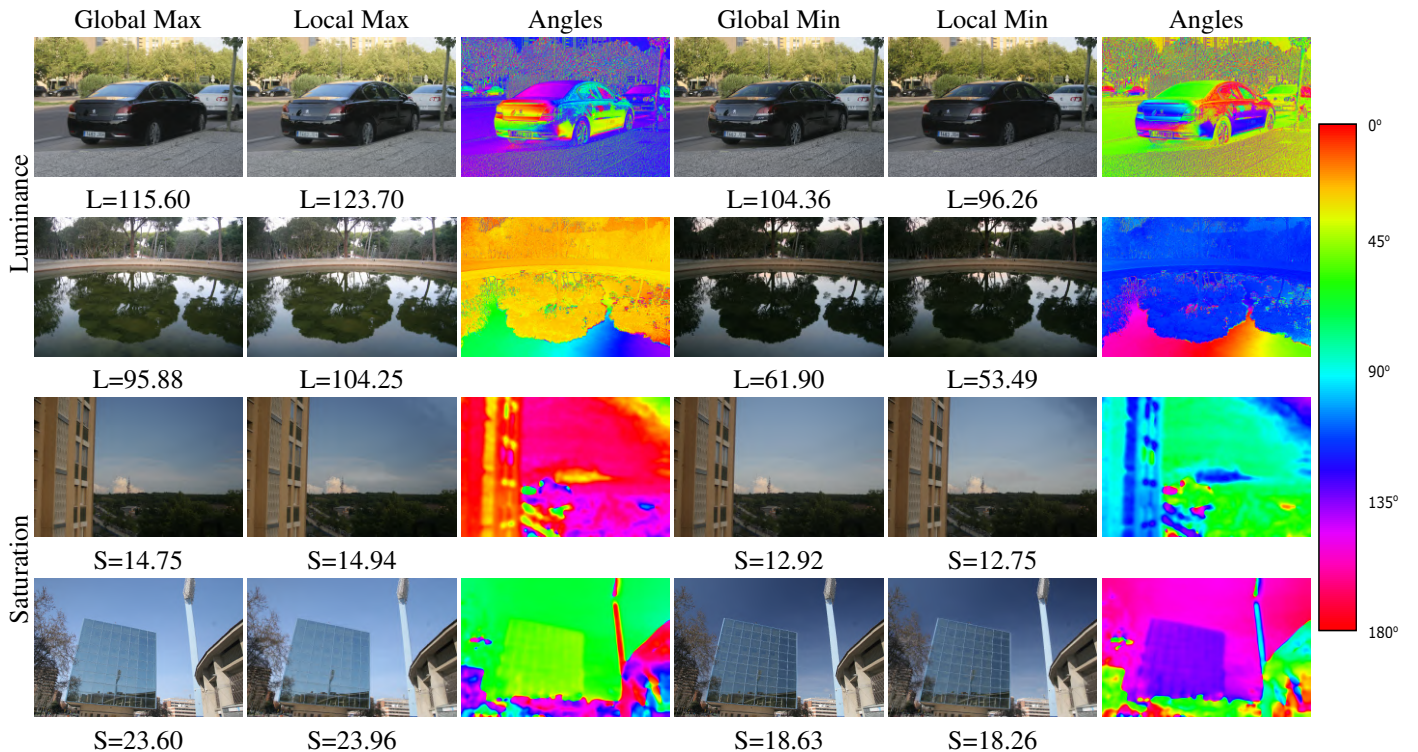

Fig. 8: Comparison between global and local filters. The first two rows are luminance optimizations while the last two rows are saturation optimization. In the first row we can see the change in the reflections of the car, the global filter focuses on the reflections of the rear window, while the local minimize and maximize also the highlights in the trunk and the side of the car. The second row shows a scene with a small pond. The local filters achieve two different things that the global filter fails at. In the maximization, the entire surface of the pond is maximized and the pond background cannot be seen while in the local minimization, at the bottom of the images the rocks at the pond floor are seen. The saturation local filters are less spectacular, and the differences between global and local are more subtle. This is because the biggest saturation change that is achieved in images with polarizing filters is seen in the sky, which in general can be almost maximized with the same angle. However, small differences can be appreciated in both scenes, as in the upper left part of the third row scene.

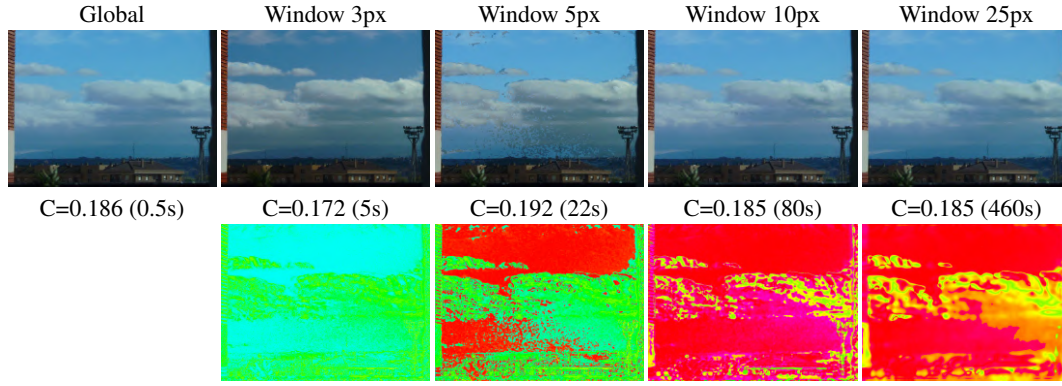| Global	Window 3px	Window 5px	Window 10px	Window 25px
C=0.186 (0.5s)	C=0.172 (5s)	C=0.192 (22s)	C=0.185 (80s)	C=0.185 (460s)

Fig. 9: Impact of different window sizes (radius in pixels above) in the local contrast filter. The computational cost increases exponentially with the size of the window and abrupt changes in the selected angle are minimized as the window size grows.

Similarly to the case of global contrast optimization, this leads to a fourth degree linear system (per pixel) with a change of variable that enables at most eight solutions, all of them tested. The issue in this case is that the size of the window $\mathbf{W_p}$ is a critical parameter: for very small windows the result can be noisy while for larger windows are slower and too similar to the global filter, as shown in Figure 9. Contrary to the luminance filter, smoothness in the angle change is not guaranteed.

Saturation. The local approach for saturation optimization is similar to the local luminance optimization: a specific angle per-pixel is selected according to the target function described in Equation (22) and the linear filter described in Equation (15). The derivative is again an order four polynomial, and the roots are also calculated with the Descartes-Euler method, per-pixel.

Unlike luminance, however, such order four derivative hints that the variation of saturation with respect to the filter angle is not smooth, hence yielding high frequency oscillations. When selecting the optimal per-pixel angle for saturation, such high frequency variations are perceived on the resulting image as high frequency noise. In order to eliminate such noise, we first calculate the optimal angle per pixel and then we apply a Gaussian blur in angle space, smoothing the angular variations between pixels (see Figure 10). While this approach deals with the high frequency noise, under some special circumstances (and kernel sizes) it can generate artifacts. Other denoising algorithms, such as non-local means, could improve the results and are interesting avenues for future work. Both success and failure cases are shown in Figure 8.

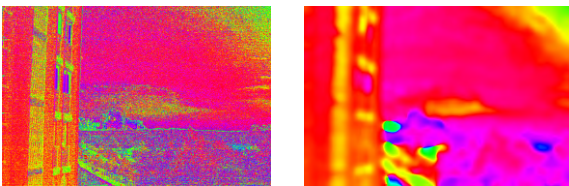More results of these three filters can be seen in our supplemental material.

Fig. 10: Angle selected in each pixel with and without blur. Left image shows the angle selected per pixel without blur, where noise and hard changes can be seen. Right image shows the result of blurring the angle of the left image, which results in smoother changes but losing the high frequency information.

5.4. Brush-based editing

The previously described polarization mapping operators, either software polarization or global and local maximization, enable interesting edits that, still, affect the whole image one way or another and can work in real time. We illustrate this real time capability with the application of such polarization mapping operators in a brush-based editing tool. We include two kinds of brushes:

- Basic software filters (as described in Section 5.1), including any commercial polarizer (both circular and linear) at any chosen angle.

- Local optimization filters (as described in Section 5.3), maximizing or minimizing luminance, contrast or saturation: allows to apply the automatic filters to maximize or minimize the parameters commented in Section 5.3.

Each brush is a circular Gaussian window, parameterized by the radius r and σ, that apply the corresponding filter to all the pixels at a distance less than r from the center. σ again is used to weigh each pixel depending on how close they are to the center. We use $\sigma = \frac{r}{3}$

The application of software filters per pixel is rather straightforward, given that it is a simple dot product per pixel affected by the brush, as described in Equation (13), and can be done in real time. In the case of local optimization of luminance, the calculation of each pixel is also fast enough for real time editing (a simple arctangent and the corresponding dot product, as shown in Equation (24)).

Both contrast and saturation filters need more complex calculus (using the Descartes-Euler method for finding the roots of a 4th degree polynomial, as described in Equation (21)). Furthermore, both require to apply such calculations to a window around each pixel. Even with parallelization, only small windows and small brushes can be used in real time. We solve this by pre-calculating the best per-pixel angle for each parameter (maximum and minimum contrast and saturation). This leads to four buffers (same size than the image) that store the optimal angle. This additional storage enables the use of all brushes in real time. Some results edited with this tool can be seen in Figure 11. Also, a video with the editing process is available as supplemental material.
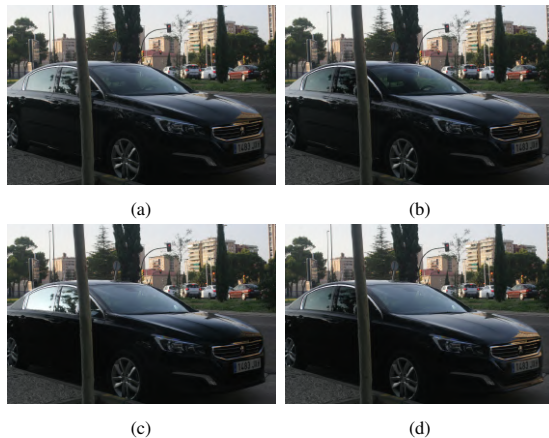
Fig. 11: (a) Image without filters (b) Minimized the luminance in the windows of the car (c) Maximized the luminance in the windows and minimized in the rest of the car (d) Minimized the luminance in the lateral of the car and maximized in the front.

Fig. 12: Left image show a translucid van produced by a moving object during the capture. Right image show blurry trees due to small movements between the different captures.

6. Discussion and future work

In this work we introduce both a simple low-cost method to capture Stokes images using only consumer hardware, and a set of tools for editing such images while taking light polarization into account. As opposed to traditional photographic pipelines, where the photographer must choose a specific polarizing filter and search for the adequate rotation of such filter before taking the picture, our software pipeline enables the search of such filters as a post-process. First, the user can explore a very wide range of polarizing filters by software. Then, software tools enable an automatic choice of the best polarizing filter for a specific goal (brightness, contrast or saturation maximization or minimization), and such choice can happen at global level (for the whole scene) or at local level (for each pixel independently). This local optimization filters can achieve results hardly obtainable with conventional photographic hardware, such as minimizing all the highlights of an image no matter their orientation.

We show an application of the polarization mapping operators in a brush-based editing tool that can apply all mentioned filters with user interaction at specific pixels, endowing the photographer with greater artistic freedom. This editing tool is a proof of concept of the real time capabilities of our filters, and future exploration of the possibilities of a more sophisticated polarization editing tool with increasing usability are interesting avenues for future work.

Our capture method generates the Stokes image from five different images with different filters (or filter rotations) of the same scene (see Section 4). Similar to HDR from multiple exposures, moving objects in dynamic scenes generate artifacts, as shown in Figure 12. In the same way that single-shot HDR cameras exist, there are prototypes for single-shot Stokes cameras [4, 19] that would solve this problem but they are still not available for consumer use.

Further avenues for future work could involve more advanced and semantic target functions for filter optimization (eg. "darken the sky"), or a more widespread exploration of polarizing filters combination when optimizing (eg. including an elliptical filter). Furthermore, industrial applications can arise from

this research: given a working Stokes camera, set up as a security camera, the local luminance minimization (which could work in real time for video feed) could reduce or eliminate specular highlights, enabling the identification of car drivers.

We hope that our approach inspires further research. For that purpose we also provide our source code and Stokes image dataset for public use [1]

References

[1] Reinhard, E, Ward, G, Pattanaik, S, Debevec, P. High Dynamic Range Imaging: Acquisition, Display, and Image-Based Lighting (The Morgan Kaufmann Series in Computer Graphics). San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.; 2005. ISBN 0125852630.

[2] Levoy, M, Hanrahan, P. Light field rendering. In: Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques. SIGGRAPH '96; New York, NY, USA: ACM. ISBN 0-89791-746-4; 1996, p. 31–42. URL: http://doi.acm.org/10.1145/237170.237199. doi:10.1145/237170.237199.

[3] Ng, R, Levoy, M, Brédif, M, Duval, G, Horowitz, M, Hanrahan, P, et al. Light field photography with a hand-held plenoptic camera. Computer Science Technical Report CSTR 2005;2(11):1–11.

[4] Tu, X, Spires, OJ, Tian, X, Brock, N, Liang, R, Pau, S. Division of amplitude rgb full-stokes camera using micro-polarizer arrays. Opt Express 2017;25(26):33160–33175. doi:10.1364/OE.25.033160.

[5] Liu, F, Cao, L, Shao, X, Han, P, Bin, X. Polarimetric dehazing utilizing spatial frequency segregation of images. Appl Opt 2015;54(27):8116–8122. doi:10.1364/AO.54.008116.

[6] Li, F, Tian, J, Tang, Y, Wang, Y. An image highlights removal method with polarization principle. In: 2015 3rd International Conference on Machinery, Materials and Information Technology Applications. Atlantis Press. ISBN 978-94-6252-120-9; 2015/11,URL: https://doi.org/icmmita-15.2015.78. doi:https://doi.org/icmmita-15.2015.78.

[7] Yang, F, Tang, X, Hu, B, Wei, R, Kong, L, Li, Y. A method of removing reflected highlight on images based on polarimetric imaging. Journal of Sensors 2016;2016.

[8] Shoji Tominaga, AK. Polarization imaging for material classification. Optical Engineering 2008;47(12):1 – 14 – 14. URL: https://doi.org/10.1117/1.3041770. doi:10.1117/1.3041770.

---

[1] http://webdiis.unizar.es/~amunoz/projects/CG2019_polarization/

[9] Ghosh, A, Chen, T, Peers, P, Wilson, CA, Debevec, P. Circularly polarized spherical illumination reflectometry. ACM Trans Graph 2010;29(6):162:1–162:12. URL: http://doi.acm.org/10.1145/1882261.1866163. doi:10.1145/1882261.1866163.

[10] Riviere, J, Reshetouski, I, Filipi, L, Ghosh, A. Polarization imaging reflectometry in the wild. ACM Trans Graph 2017;36(6):206:1–206:14. URL: http://doi.acm.org/10.1145/3130800.3130894. doi:10.1145/3130800.3130894.

[11] Toisoul, A, Dhillon, DS, Ghosh, A. Acquiring spatially varying appearance of printed holographic surfaces. In: SIGGRAPH Asia 2018 Technical Papers. ACM; 2018, p. 272.

[12] Rahmann, S, Canterakis, N. Reconstruction of specular surfaces using polarization imaging. In: CVPR. 2001,.

[13] Huynh, CP, Robles-Kelly, A, Hancock, ER. Shape and refractive index from single-view spectro-polarimetric images. International Journal of Computer Vision 2012;101:64–94.

[14] Atkinson, GA, Hancock, ER. Recovery of surface orientation from diffuse polarization. IEEE Transactions on Image Processing 2006;15(6):1653–1664. doi:10.1109/TIP.2006.871114.

[15] Osante, D, Muñoz, A. Fotografía computacional para el filtrado de la polarización en fase de post-proceso. Master thesis; Universidad de Zaragoza; 2013.

[16] Bromley, BC, Melia, F, Liu, S. Polarimetric imaging of the massive black hole at the galactic center. The Astrophysical Journal 2001;555(2):L83–L86. doi:10.1086/322862.

[17] Novikova, T, Pierangelo, A, De Martino, A, Benali, A, Validire, P. Polarimetric imaging for cancer diagnosis and staging. Optics and photonics news 2012;23(10):26–33.

[18] North, JA, Duggin, MJ. Stokes vector imaging of the polarized skydome. Appl Opt 1997;36(3):723–730. URL: http://ao.osa.org/abstract.cfm?URI=ao-36-3-723. doi:10.1364/AO.36.000723.

[19] M. Vedel S. Breugnot, NL. Full stokes polarization imaging camera. Polarization Science and Remote Sensing V 2011;8160. URL: https://doi.org/10.1117/12.892491. doi:10.1117/12.892491.

[20] Shurcliff, WA. Polarized light. production and use. Cambridge, Mass: Harvard University Press,— c1966 1966;.

[21] Hecht, E, Zajac, A. Optics, chapter 8. 2002.

[22] Schaefer, B, Collett, E, Smyth, R, Barrett, D, Fraher, B. Measuring the stokes polarization parameters. American Journal of Physics 2007;75(2):163–168.

[23] Collett, E. The description of polarization in classical physics. American Journal of Physics 1968;36(8):713–725.

[24] Euler, L. Of a new method of resolving equations of the fourth degree. In: Elements of Algebra. Springer; 1972, p. 282–288.