

Chapter 1

Methodology

This chapter contains the research methodology employed while implementing a proof-of-concept of an adaptive shadow removal model. Our methodology is divided into two components: algorithm assessment in the context of shadow removal in diverse environments, and the creation of the adaptive model for a previously assessed algorithm.

We first assess the previously catalogued algorithms (Chromacity, Physical, Geometry, SRT, and LRT removal) for sensitivity to both environmental and parametric change. We then perform a similar sensitivity analysis upon the parameters utilized by the algorithms. These assessments are performed qualitatively to determine the feasibility of an adaptive model being created for each algorithm.

From the sensitivity assessments, we demonstrate the creation of an adaptive model for a previously analyzed algorithm parameter. The process of

creating an adaptive model, outlined in section ??, is then detailed. We consider various environmental properties and evaluate their correlations to the selected algorithm parameter. Potential indirect correlative factors are detailed in sections ?? and ?. Finally, methodology concerning assembling the adaptive model follows in section ??.

1.1 Algorithm Assessment

We assess several leading shadow removal algorithms (Chromacity, Physical, Geometry, SRT, and LRT removal) for their sensitivity to different environmental properties, as well as quantitative shifts in those environmental properties. Environmental properties are identified across diverse datasets, examples including shadow strength/darkness, shadow directionality, and color saturation. Environmental property shifts are characterized in multiple datasets, i.e., strong illumination changes within one dataset sequence.

In addition to analyzing the algorithms' sensitivity to environmental change, we similarly assess the algorithms' sensitivity to parametric shift, i.e., algorithm performance is qualitatively assessed according to the algorithm's dependence on one or more curated parameter values. Analysis and assessment of the previously presented shadow removal algorithms is conducted using a series of graphical tools in conjunction with ground truths identifying shadow regions across eight different datasets with seven unique environments.

1.1.1 Data Collection

This section details the datasets used to conduct our assessment, as well as tools developed for analyzing the sensitivity of algorithms and parameters. The tools and techniques covered allow for both qualitative and quantitative assessment.

Datasets

A total of eight datasets were chosen for this assessment, and are summarized in Table ???. Datasets 2 - 8 (denoted by the prefix "aton_") are provided under the Computer Vision and Robotics Research Laboratory (CVRR) in association with the University of San Diego [??]. The datasets were cataloged with the Autonomous Agents for On-Scene Networked Incident Management (ATON) project [??]. These datasets represent diverse environments, including both indoor and outdoor scenes, a range of shadow intensity, and images of varying saturation.

In addition, two datasets were included from the Performance Evaluation of Tracking and Surveillance (PETS) project [??]. In contrast to the ATON datasets, these sequences were explicitly chosen because they possess subsequences featuring rapid illumination changes, due to a shift in weather conditions and cloud cover. These environmental changes darken, lighten, and blend shadows in accordance with a background model. Figure ?? demonstrates the illumination changes within the PETS datasets.

All eight datasets were manually segmented to form ground truths that

Name	Resolution	# Ground Truth Frames
PETS1	768x576	12
PETS2	768x576	11
aton_highway1	320x240	8
aton_highway3	320x240	7
aton_room	320x240	22
aton_campus	320x240	53
aton_hallway	320x240	13
aton_lab	320x240	14

Table 1.1: Dataset information.

identify shadow regions (gray) within foreground objects (white). Figure ?? illustrates the ground truths of dataset frames.

The .avi sequences aren't hosted where they were, meaning I probably can not find the sequence length.

Assessment Tools

The first stage of our methodology is an exploration of the sensitivity of shadow removal algorithms to environmental and parametric change. We developed tools to assist intuition regarding an algorithm's performance, capable of allowing a user to manually tune an algorithm parameter, and observe its effect on performance.

To better understand an algorithm's dependence on a parameter, a framework was required to rapidly modify the parameter in question. The initial implementation of the shadow removal methods, courtesy of Sanin et al. [??], contained relevant parameters hard-coded into the algorithms. Each

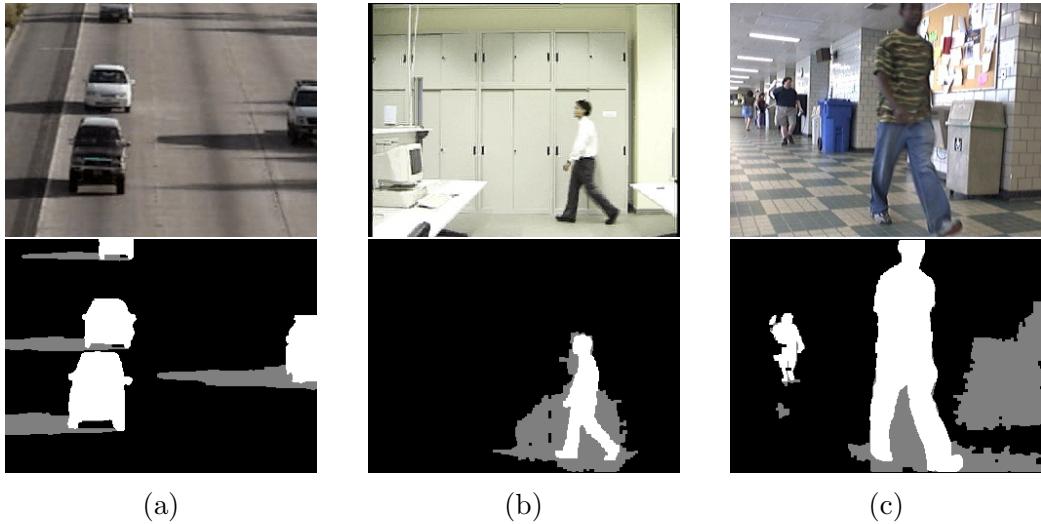


Figure 1.1: Datasets collected for ATON: (a) aton_highway1, (b) aton_lab, and (c) aton_hallway.



Figure 1.2: PETS2 experiencing both high illumination (a) and low illumination (b) (due to cloud cover).

```

56 [PhysicalShadRemParams]
57 coneAngle = 0.15707963267
58 coneR1 = 0.3
59 coneR2 = 1.0
60 weightSmoothTerm = 4
61 learnBorders = 1
62 gmmGaussians = 5
63 gmmInitVar = 30
64 gmmMinVar = 1
65 gmmStdThreshold = 2
66 gmmWinnerTakesAll = 0
67 gmmLearningRate = 0.1
68

```

Figure 1.3: An example SimpleINI file. Parameters taken from this file are used to adjust values in real-time.

algorithm therefore required re-compilation to bring any modifications to fruition. This shortcoming was rectified by extracting each parameter and organizing them into an `.ini` file. By utilizing [username brofield](#)'s SimpleINI architecture, hosted on Github [??], parameters can be either set extemporaneously or altered by an external script. An example SimpleIni `.ini` file is shown in Figure ??.

[Graphical tools \(GUI\)](#) were then developed to rapidly assess and visualize a parameter's affect on shadow removal. Each shadow removal method was modified to accept arbitrary parameter values in real-time from the `.ini` file. These parameters were then tied to graphical sliders (from OpenCV's `highgui` library [??]) dictating their range and value. Any numerical parameter that has the potential to modify the Detection or Discrimination rates of a shadow removal method is included in the GUI. An example of the GUI is seen in Figure ??.

The process in which the GUI modifies and displays

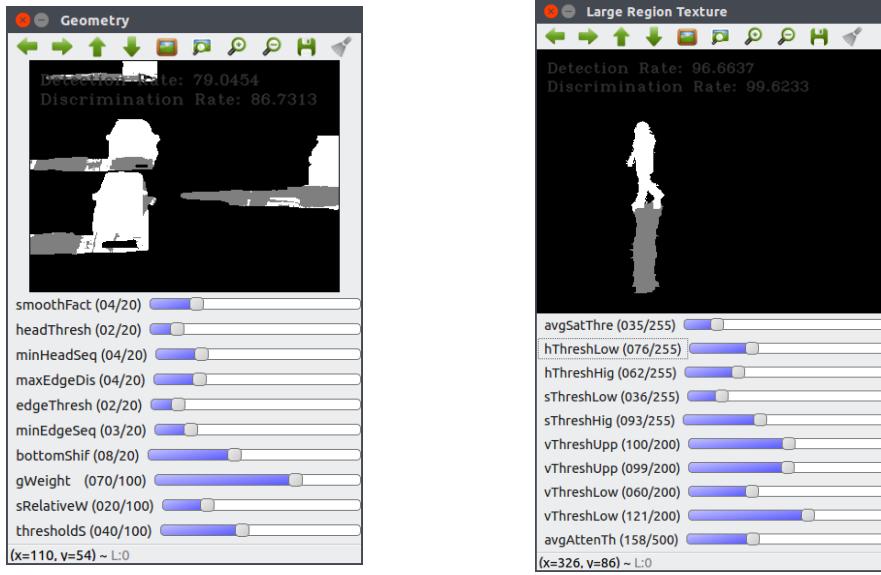


Figure 1.4: GUI tools created using OpenCV, displaying Geometry removal and LRT removal.

parameters is illustrated in Figure ??.

The display is updated in real-time with both a visual representation of detected shadow pixels, and a quantified display of both the exact detection and discrimination rates. Leveraging human perception (qualitative perception?), the graphical tools enabled rapid evaluation of the sensitivity of an algorithm to changes in parameter values within specific scenes.

Determining Optimal Parameter Values

The removal of compilation enables us to create batch jobs to systematically vary parameter values across multiple runs. This allows us to meticulously exercise a given parameter (pr) through a specified range and record its corre-

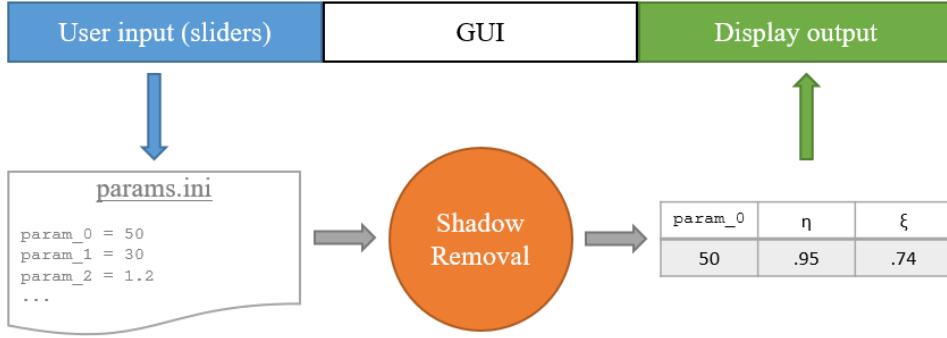


Figure 1.5: The GUI takes user input through sliders, updates values in a .ini file, which are used to produce new detection and discrimination rates. The GUI then visualizes this output.

sponding affect upon detection and discrimination rates. Figure ?? illustrates the data collection process. The calculated detection and discrimination responses to change in pr are recorded as the functions $\eta(pr)$ and $\xi(pr)$. Both $\eta(pr)$ and $\xi(pr)$ are determined for one frame. Figure ??(a) demonstrates these responses.

We also utilize the iterative process to determine the optimal value of pr for each frame in a dataset, i.e., which pr value yields the greatest improvements in detection and discrimination. This optimal value is denoted as pr^* . The detection and discrimination rates calculated are summed, resulting in a shadow removal score $\Xi(pr)$ (Eqn. ??), which quantifies the efficacy of the shadow removal algorithm given a certain pr value. Figure ??(b) illustrates the score calculated using the responses found in Figure ??(a). We define the optimal value pr^* as the global maximum of $\Xi(pr)$ (Eqn. ??).

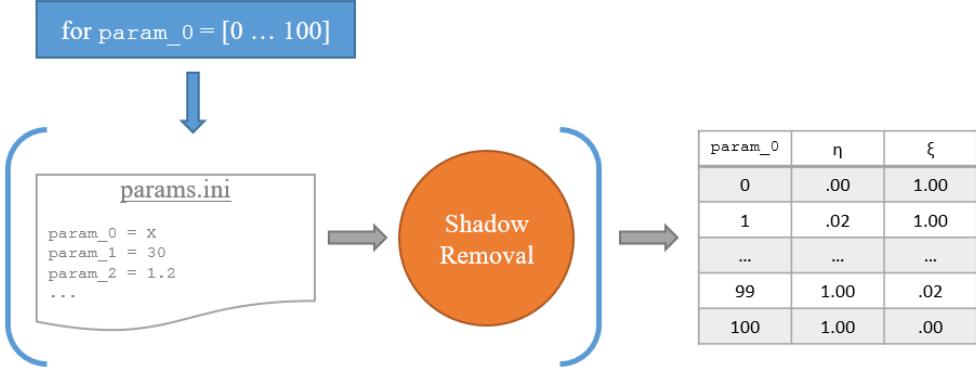


Figure 1.6: For one frame, a parameter is systematically iterated to provide detection/discrimination results for each possible parameter value.

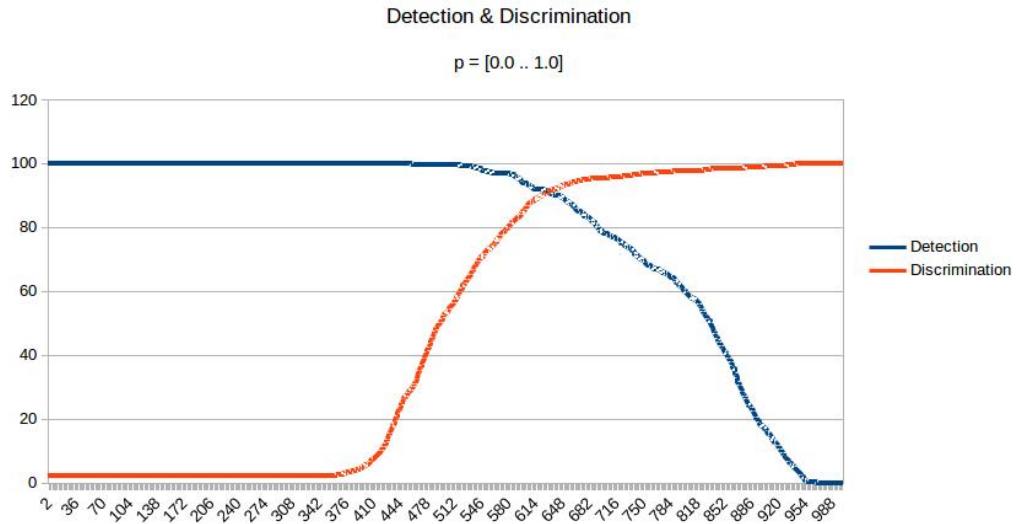
$$\Xi(pr) = \eta(pr) + \xi(pr) \quad (1.1)$$

$$pr^* = \Xi^*(pr) \quad (1.2)$$

To assess the performance of algorithms throughout a dataset, pr^* is calculated for each frame. The pr^* values are used to determine degrees of correlation for an environment, and set the foundation for a general model of arbitrary shadow removal improvement.

1.1.2 Algorithm Selection Strategy

Using the tools detailed in section ??, we perform a brief qualitative assessment of the shadow removal algorithms' sensitivity. Given the diversity of shadow removal methods, a wide array of environmental factors potentially



(a) Detection (blue) and Discrimination (orange) rates are charted against the iterated parameter value (x-axis).



(b) For each parameter value, detection and discrimination are summed to produce a score. The global maximum of this value is the optimal parameter value (pr^*).

Figure 1.7

may influence an even wider array of algorithmic parameters. The scope of this research has therefore been focused on identifying a suitable algorithm to assess, linking a parameter (intrinsic to this algorithm) to unique environmental properties or changes, and developing a model for real-time parameter adaptation based on correlations found.

Candidate shadow removal algorithms are evaluated according to the correlation between their detection and discrimination over time, and qualitative observations attributed to a dataset; e.g., the PETS1 dataset experiences a large illumination change midway through its sampling, which is taken into account when searching for trends among the accuracy of a candidate method. For example, Figure ?? charts the efficacy of each shadow removal method against the number of SIFT parameters detected in a frame.

Evaluation of Methods

We evaluate popular shadow removal methods introduced in section ??: Chromacity, Physical, Geometry, SRT, and LRT. This section describes our rationale for using Physical shadow removal as part of our proof-of-concept, implemented starting in section ???. Our evaluation considers the tunable parameters of an algorithm, dependence on environmental properties and content, and sensitivity of relevant parameters.

A parameter's sensitivity is evaluated by viewing its detection and discrimination responses ($\eta(pr)$, $\xi(pr)$). Figure ?? illustrates what is considered a sensitive parameter. A parameter is considered sensitive when a narrow

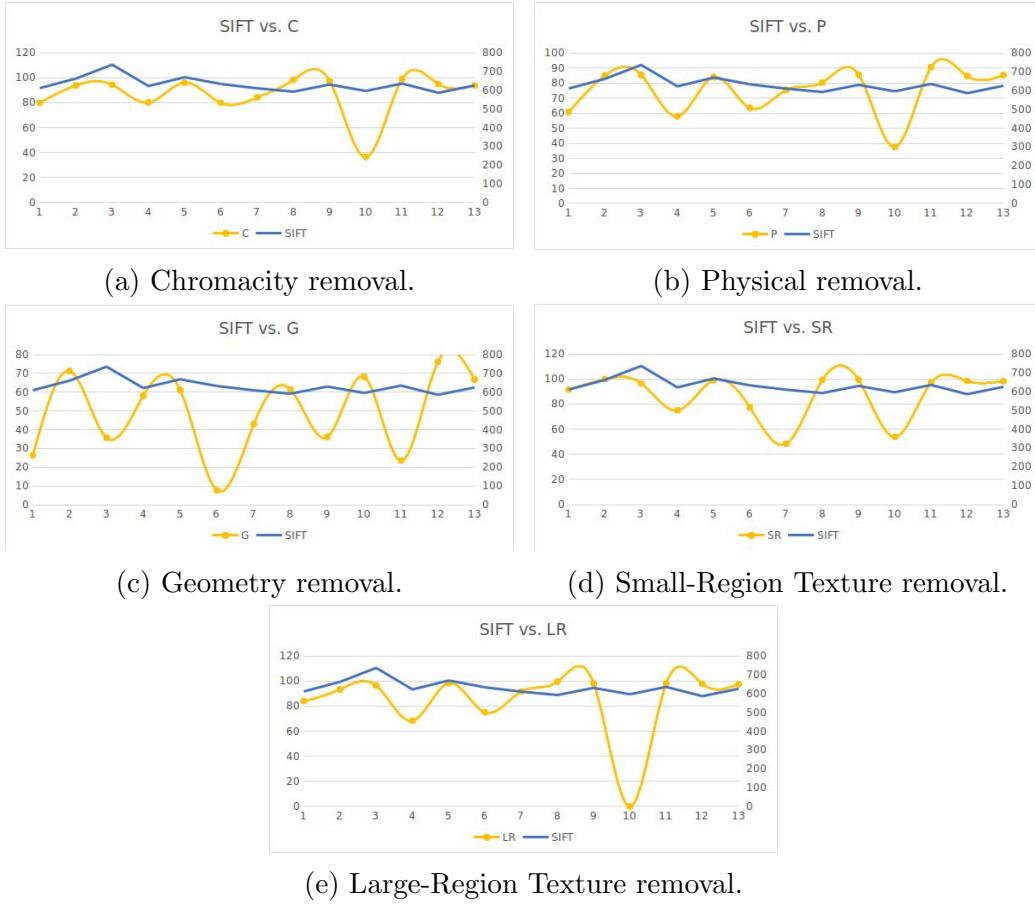


Figure 1.8: Shadow removal methods and their performance v. number of detected SIFT features in an aton_hallway.

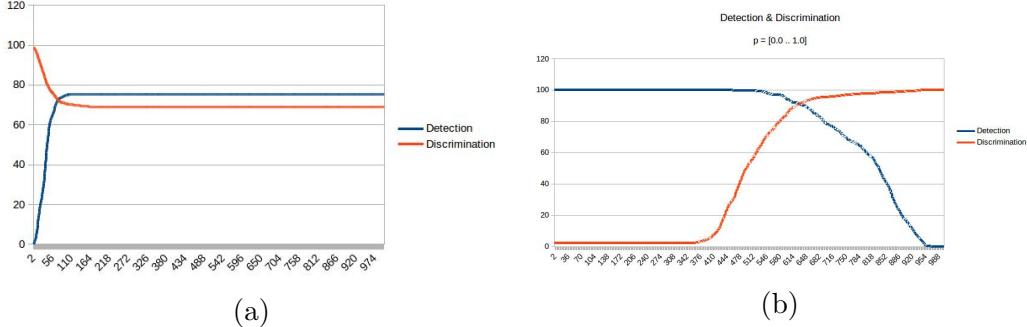


Figure 1.9: (a) depicts a sensitive parameter, while (b) represents a typical parameter.

range of parameter values affect detection and discrimination rates disproportionately. This sensitivity causes problems for an adaptive model, as the range of potential optimal parameter values (pr^*) is too small to correlate with environmental properties.

Both Geometry removal and SRT removal were shown to behave highly erratically from frame to frame even within an environmentally consistent dataset, demonstrated in Figure ???. Geometry-based removal remains dependent on the shape and consistency of processed foreground objects, and therefore dependent on the consistency of whichever foreground extractor is used for a scene. Furthermore, the algorithmic parameters associated with Geometry removal apply only to scenes in which the previous dependency is already fulfilled, i.e., the tunable parameters are relevant only to the geometric shapes necessary to enable Geometry removal. These dependencies are not related to the properties of shadows within an environment, therefore we do not pursue further analysis with Geometry removal. SRT removal is

based upon a series of Gabor filters used to characterize textural patches found in shadows, and match them to a corresponding background model. This technique requires that shadows cover textural portions of the background model large enough to perform meaningful analysis. Manual tuning of SRT’s algorithmic parameters provides no benefit to the algorithms detection or discrimination rates. This dependence on a shadow’s area, coupled with the lack of parameters suitable to improve shadow removal, eliminated SRT removal from candidacy.

Sanin et al.’s Large-Region Texture algorithm was also eliminated from the assessment due to its inconsistent shadow removal, similar to Geometry and SRT removal (Figure ??). Some environments, such as the roadways of aton_highway1, yielded null detections for all frames. The Large-Region Texture algorithm recognizes the pitfalls of the Small-Region approach, such as restrictive texture regions, and attempts to correct them using hard-coded parameters. The algorithm primarily displays a sensitivity to environment, as identical parameter values produce vastly disparate shadow removal performances (Figure ??). LRT also demonstrates varying levels of parametric sensitivity, also dependent on the deployment environment. For example, in environments such as aton_campus and aton_hallway, the parameter *avgAttenThresh* is highly sensitive, similar to Figure ??(a); however, when deployed in aton_highway1, *avgAttenThresh* does not affect detection or discrimination for any value.

Chromacity and Physical shadow removal provide tunable parameters

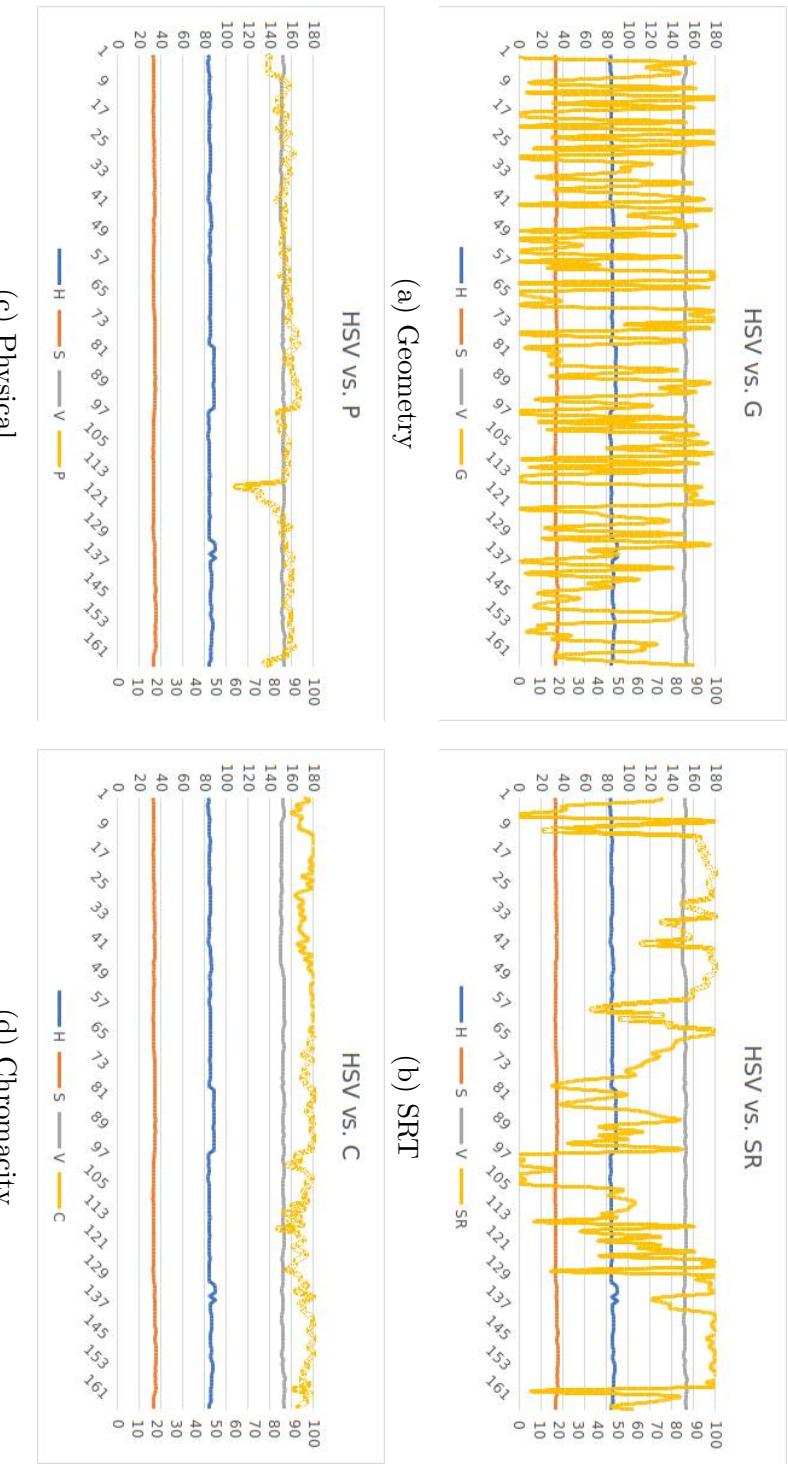
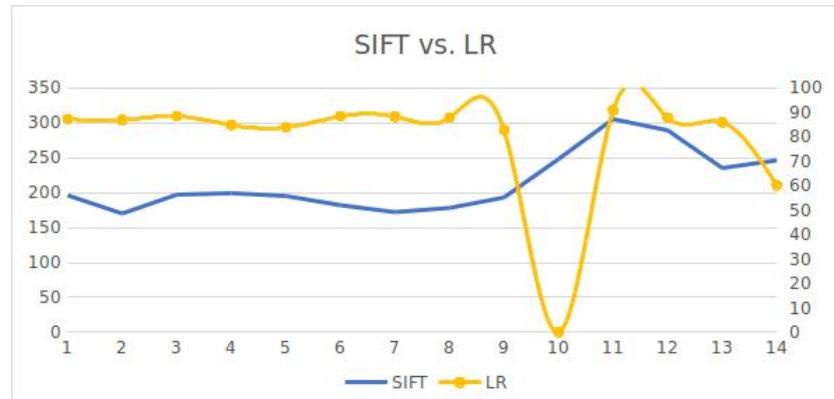
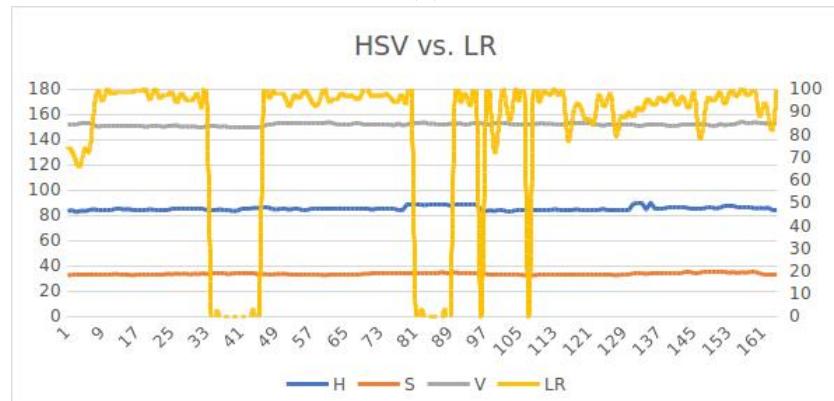


Figure 1.10: (a) and (b) showcase erratic shadow detection, compared to (c) and (d) using the same frames.



(a)



(b)

Figure 1.11: (a) displays the consistent nature of LRT in some environments, while (b) demonstrates erratic dips in consistency.

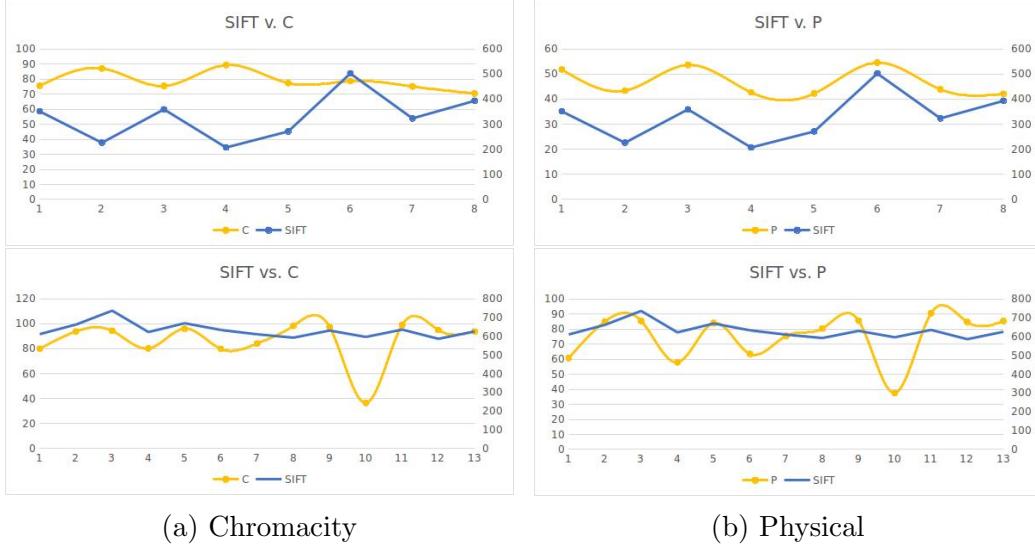


Figure 1.12: Chromacity (a) and Physical (b) shadow removal demonstrate similar responses to multiple datasets.

that lie within acceptable ranges of sensitivity. An example of an acceptable range of sensitivity is seen in Figure ??(b). Chromacity and Physical shadow removal also display consistency between datasets, i.e., none of the datasets provided particularly poor results that are not linked to an environmental property change, such as illumination change. The Chromacity removal algorithm succeeds due to the tendency of shadows (within a dataset) to remain a consistent darkness when compared to the corresponding background model. However, this simple approach can prove problematic when illumination changes occur within a dataset. Physical shadow removal utilizes a similar system of gauging a shadow's darkness compared to a background model - what is referred to as a weak detector [??] - and implicitly suffers similar breakdowns of detection when presented with significant illumination

flux within a dataset. Although they both suffer, Physical shadow removal employs a strong detector [??] to increase detection and discrimination at the cost of processing time. In the end, Physical shadow removal was selected as the main experimental framework for this study, triumphing over Chromacity shadow removal in part due to its more sophisticated model.

1.1.3 Selecting a Parameter - Physical Shadow Removal

Physical shadow removal operates in two stages: the weak detector, and the strong detector. The weak detector typically identifies candidate shadow pixels simply by eliminating impossible pixels, i.e., pixels that are brighter than the background model. These candidate pixels are then provided to the strong detector, which characterizes these pixels as either shadow or foreground. Our assessment of algorithmic parameters explores parameters in both of these detector spaces. Before we can illustrate which parameters were considered for experimentation, we must first understand the parameters' place within the algorithm.

Weak Detector - Physical Shadow Removal

The purpose of the weak detector is to eliminate impossible pixels from being fed to the strong detector. The weak detector in Physical shadow removal functions similarly to the Chromacity shadow removal process; the candidate

pixel is evaluated by its distance from a corresponding background pixel and sorted accordingly. The weak detector is visually, and technically, a cone projected in an RGB plane indicating the range in which a normalized shadow pixel may lie **with** respect to a background model. Normalizing the shadow pixel's position relative to the background model is accomplished by computing the angular distance between the RGB vector of the foreground and the RGB vector of the background. The resultant represents the color-space deviation from foreground to background. This threshold parameter is named *coneAngle* within the Physical shadow removal algorithm. The cone represents two dimensions: color deviation, and intensity (or brightness) deviation.

A foreground pixel is considered a possible shadow if the normalized ratio of foreground to background brightness (scaled by a function of the color deviation) is within a pre-defined brightness range. The tolerable brightness range, delimited by parameters named *coneR1* (minimum) and *coneR2* (maximum), is hard-coded within the original algorithm to be 0.3 to 1.0. This range is the maximum and minimum that the normalized ratio of foreground to background (multiplied against the cosine of the color deviation) can exist within and still be considered a possible shadow (awkward). [??]

illustrates the conic model in Figure ??.

The main components of the weak detector are parameterized within the algorithm to fit a general model; $coneR1 < x < coneR2$ as an intensity range works reasonably well for many scenes, but there exists an optimal range for any given environment. Similarly, the maximal color angle between fore-

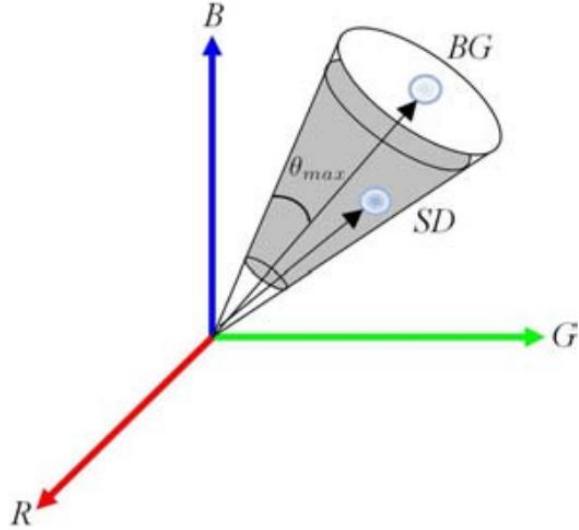


Figure 1.13: The conical volume describing the tolerable brightness range of a shadow pixel in relation to its illuminated counterpart.

ground and background vectors is dependent on environmental parameters. These parameters, *coneR1*, *coneR2*, and *coneAngle* are therefore considered when looking for correlations with environmental conditions.

Strong Detector - Physical Shadow Removal

After the weak detector removes impossible candidates for shadow pixels, a Gaussian Mixture Model (GMM) is used to learn the color features of shadow pixels when compared to background pixels. Using the remaining pixels, the GMM estimates the normalized spectral ratio for shadows in a scene, i.e., the ratio of spectral illuminants [??, ??]. More information on spectral illuminants can be found in section ?? . The GMM is a framework based upon Expectation Maximization and learning over time; therefore, the initial pa-

rameters presented in the algorithm have little influence over the efficacy of shadow removal. Shadow removal displayed sensitivity to only one significant parameter, *postThresh*, the parameter controlling the posterior threshold. The posterior threshold is the threshold governing shadow/foreground assignment after the posterior probability of a pixel is determined via the GMM. Since the GMM adapts to its environment, modifying the posterior threshold invalidates any learning the GMM has achieved. Correlating *postThresh* to environmental parameters is not considered in this study.

Evaluation of Parameters

Due to their integral nature to both the weak and strong detectors, the following parameters are evaluated for their effect on shadow removal: *coneR1*, and *coneAngle*. Each parameter is shown to have a pronounced effect on shadow removal. Figure ?? demonstrates *coneR1*'s contribution to detection/discrimination across datasets.

coneR1's effect on shadow removal remains consistent and scalable. The parameter is exercised through a large range, with an equally large range of resultant detection and discrimination rates. The wider range and relatively coarse-grained nature of the parameter allows for an easier time identifying trends and correlations with the iterations.

coneR2 represents the upper limit of the cone around a background pixel's color representation. The parameters are inextricably tied; however, *coneR2* was not subjected to as rigorous examination, as *coneR2*'s dependence on

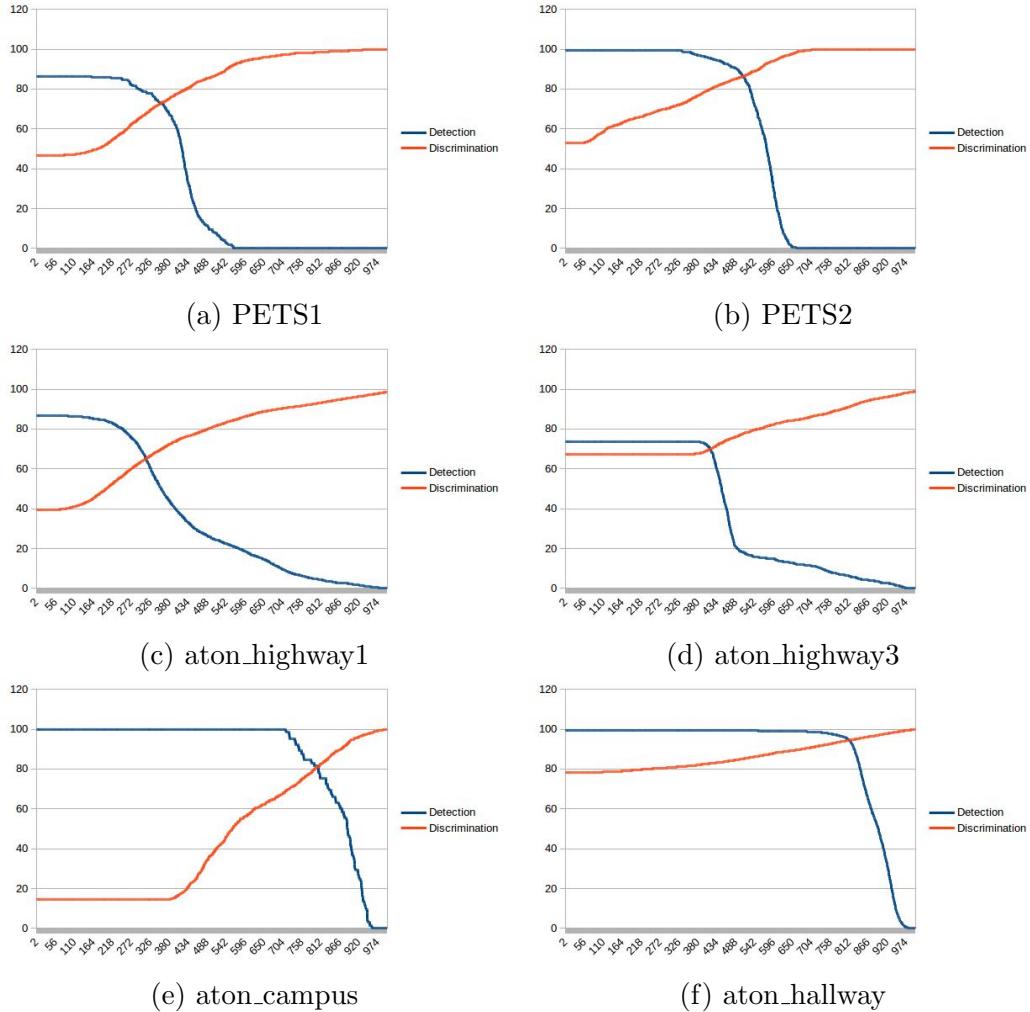


Figure 1.14: Detection (blue) and discrimination (orange) rates are calculated after iteration of *coneR1* (x-axis). Full results found in appendices.

the value of *coneR1* creates an exponential field of possible values for each variable.

The remaining parameters (results included in appendices) inhabit too narrow a range of sensitivity to properly exploit, e.g., *coneAngle* displays erratic variation in the discovered values for optimal shadow removal, yet the optimal values are within millionths of each other. Similarly, the remaining parameters do not provide as large an improvement in shadow removal as *coneR1* does. The erratic nature of the optimal parameter values, in combination with both the minimal removal improvement and **narrow** range of sensitive values makes further examination of these parameters difficult. For the duration of the study, environmental feature correlations were sought after only in relation to *coneR1*.

1.2 Environmental Assessment - Environmental Properties

We now create a proof-of-concept of our adaptive model, by correlating environmental properties with a previously assessed parameter *coneR1*. We explore observed environmental properties within the datasets: attenuation and saturation. We also perform sensitivity analysis on multiple environmental properties (brightness functions, SIFT parameters) to determine the extent of their possible influence on correlation between *coneR1** and attenuation (α). A model is detailed for using the observed environmental properties to

calculate a new variable $coneR1'$, which represents the adaptation of the $coneR1$ parameter to more closely match the curated $coneR1^*$ ideal value.

1.2.1 Previous Work - Large Region Texture Removal

Of all the leading shadow removal algorithms we evaluated, only one attempts to adapt its parameters to its current environment. In Sanin et al.'s LRT shadow removal, there are three examples of ambient environment properties taken into account: average saturation (of foreground pixels), average attenuation (from foreground to background), and average perimeter size of foreground objects. These three global properties govern the value of certain algorithm parameters.

LRT removal checks the average perimeter of foreground objects in a frame against a predetermined threshold. Average perimeter is linked to three additional algorithm parameters that control the size of the texture region being matched to identify shadows. If the average perimeter is above the predetermined threshold, the range of area for foreground/background correlation is expanded. The average perimeter is computed on a per-frame basis, while the threshold remains static.

LRT removal, like Physical shadow removal, uses a weak detector to retrieve candidate (shadow) pixels. The candidate shadow pixels must fit within a certain range of Hue (H), Saturation (S), and Value (V)(or Brightness) when compared against the background model, i.e., the foreground HSV values subtracted from the corresponding background HSV values, referred to

as ΔHSV , must fall within a specified range. There are two definitions of this acceptable range: $\Delta\text{HSV} \in \{[0, 76], [0, 36], [0.6, 1.0]\}$, and $\Delta\text{HSV} \in \{[0, 62], [0, 93], [0.21, 0.99]\}$. If the average saturation of all identified foreground pixels exceeds a predetermined threshold (average saturation = 35), the higher values of H and S are used as the acceptable range. Similarly, if the average attenuation of the foreground pixels exceeds its threshold (average attenuation = 1.58), the latter range of [0.21, 0.99] is used for evaluating brightness.

This need for adaptability underscores the motivation of this research to discover a scalable and portable solution for parameterization. In the case of Physical shadow removal, foreground object perimeter has no bearing on the accuracy of shadow removal, due to Physical removal utilizes a wholly pixel-based approach. However, both average saturation of foreground objects and average attenuation were further examined in this study.

1.2.2 Attenuation and Saturation

After exploring previous work performed in section ??, we focus our research on evaluating the attenuation and saturation of an environment. In this section, we define attenuation and saturation.

Attenuation (α) represents the loss of intensity from foreground pixel to background pixel in a frame. Traditionally, brightness attenuation is represented mathematically in decibels (dB) as the ratio of background and foreground brightness (Eqn. ??). The *brightness()* function used here is the HSV brightness function (Eqn. ??), which defines brightness as the maxi-

mum value between the R,G,B channels of a pixel.

$$\alpha_{dB} = \frac{brightness(\vec{bg}(p))}{brightness(\vec{fg}(p))} \quad (1.3)$$

p represents pixel coordinates. In an effort to better capture shadow pixels, we discard any pixel where $\alpha < 1.0$, implying it is brighter than its corresponding background. While the α_{dB} representation adheres most closely to the physical definition of light attenuation within a shadow, we consider the reciprocal of this model ($1/dB$) of attenuation for correlation. As all considered pixels have an α_{dB} of at least 1.0, the reciprocal bounds the value between 0.0 and 1.0. This range is more suitable for correlation calculations, due to the covariance between data points being easily affected by large changes in value. More information regarding correlation is found in section ??.

Alternatively, we consider the *percentage change* ($\% \Delta$) model of light attenuation. This model is defined in Eqn. ??.

$$\alpha_{\% \Delta} = \frac{brightness(\vec{bg}(p) - \vec{fg}(p))}{brightness(\vec{bg}(p))} \quad (1.4)$$

$(1 - \% \Delta)$ is considered for correlation, which indicates the percentage of a background pixel's intensity the foreground represents.

Saturation is the measurement of 'depth of hue' in a pixel, e.g., a lower saturation means the base hue is less expressed, while a higher saturation means the color of a pixel more closely matches the base hue. The saturation

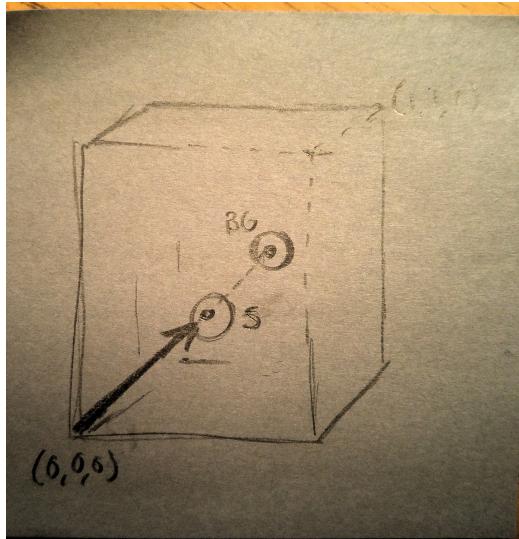


Figure 1.15: Traditional understanding of brightness attenuation assume a shadow pixel must lie between its illuminated background model, and the origin in the RGB space.

level of a background pixel affects the intensity attenuation a shadow pixel experiences. Saturation properties are discussed in greater detail in section ??.

1.2.3 Non-linear Attenuation and Spectral Properties of Light

In the ideal case, the attenuation of light is an entirely linear function, i.e., a shadow pixel lies along the the vector drawn from the background pixel to the origin within the three dimensional RGB space. [re Physical]

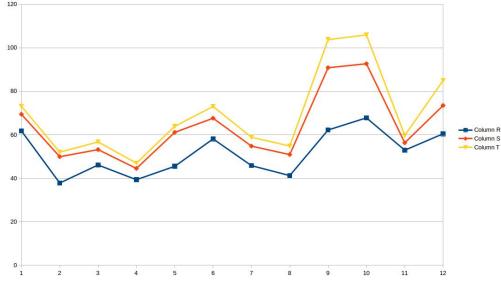
Due to physical properties of illumination and reflectance, this linearity does not always hold true; cast shadows in real-world scenes are caused by

more than one illumination source. In an outdoor scene, a shadow may be the product of direct sunlight, blue light refracted from the sky, and diffuse light scattering from nearby objects or surfaces. Any of these factors contribute to color bleed [??], influencing the attenuation of shadow pixels. These disparate light sources are said to have different *spectral power distributions*, different illumination characterized by varying concentrations of constituent wavelengths. This non-linear attenuation model is the primary motivator behind using GMM to learn a shadow’s color model in [??].

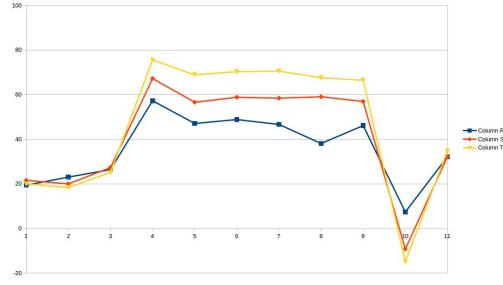
Observed Spectral Properties in Outdoor Scenes

Since the spectral properties of an illumination source cannot always be predicted, this study **attempts** to observe trends **across** datasets to properly characterize shadows. By visualizing the magnitude of color shift due to shadows, we can clearly partition datasets into outdoor and indoor images. Outdoor datasets experience change disproportionately in each channel of an RGB image, while indoor datasets are characterized by equal and predictable shifts in each channel.

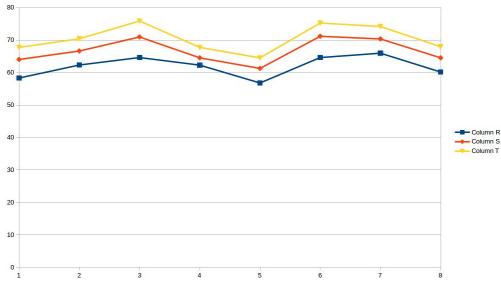
This disproportionate discrepancy is most apparent during periods of large illumination change, found prominently in datasets PETS1 and PETS2. Examining the outdoor datasets, we can see that the red and green channels experience larger magnitude shifts **than** in the blue channel. Based on assumed spectral power distributions of the source illumination found in the outdoor scenes, we can deduce that outdoor scenes display an introduction of



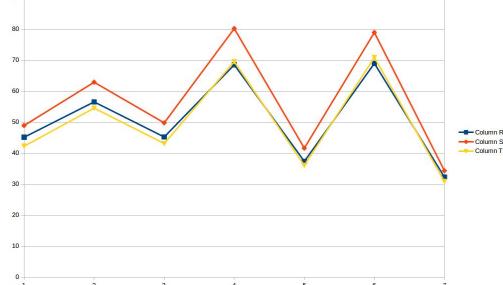
(a) PETS1



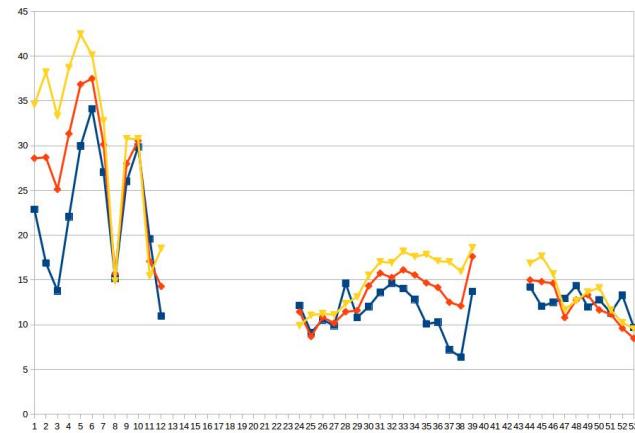
(b) PETS2



(c) aton_highway1



(d) aton_highway3



(e) aton_campus

Figure 1.16: Outdoor datasets demonstrate consistently greater deviations in the red and green channels. During illumination changes (evident in PETS1 and PETS2), the red and green channels shift disproportionate to that of the blue channel, indicating scattered blue light is a primary component of the shadows' spectral illuminant ratio.

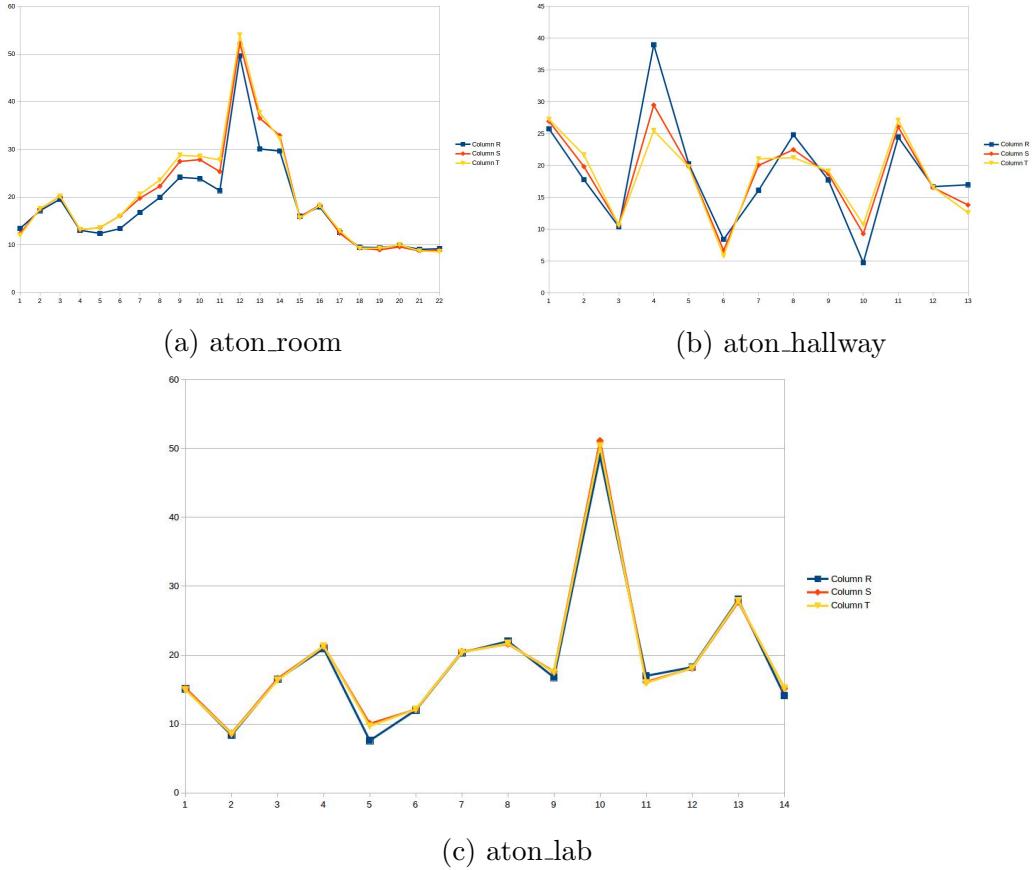


Figure 1.17: Indoor datasets demonstrate closer grouping of each channel's color shift. aton_hallway behaves most erratically of the indoor datasets. This is due to color-bleed from nearby objects, making aton_hallway the most diverse spectral ratio of the indoor sets. aton_lab behaves linearly, as expected.

blue light scattered from the sky that is not present in indoor scenes. With a spectral power distribution skewed towards blue, the red and green channels experience greater perturbation than the blue channel due to cast shadows. This study attempts to utilize this observed multi-illuminant model to better characterize the attenuation model assumed by a scene.

1.2.4 Brightness Models

Since attenuation is a function of brightness, we attempt to utilize multiple methods of calculating brightness to improve attenuation modeling. We use different brightness functions in the calculation of α for each dataset, hypothesizing that correlation between α and *coneR1** can be improved using different brightness functions. We conduct a sensitivity analysis to determine the effect that different brightness models can have on attenuation.

Below is a short taxonomy of popular brightness models studied. Each of the brightness models included are tested and analyzed for its affect on the α model.

HSV

Described as a 'hexcone' model, [??], brightness in an HSV representation is defined as the maximum value of the pixel represented in the red, green, and blue channels ($R\vec{G}B(p)$).

$$V = \max(R, G, B) \quad (1.5)$$

HSV can lead to misrepresentation of brightness as it is perceived by the human eye. This is because, as noted in Rec. 709/601 [ref 709], the same intensity of green appears brighter than that of red, which, in turn, appears brighter than the same intensity of blue. In the HSV scheme, green at full intensity ($R\hat{G}B = (0, 1, 0)$) matches the brightness of blue at full intensity ($R\hat{G}B = (0, 0, 1)$). It also minimizes contributions from channels other than the dominant channel, e.g., $R\hat{G}B(0, 1, 0) = R\hat{G}B(.9, 1, .9)$. HSV therefore is best suited for environments characterized by abnormally low saturation, where light attenuation due to shadows is most linear.

HSI

HSI represents the most basic understanding of brightness, as *Intensity*.

$$I = \frac{R + G + B}{3} \quad (1.6)$$

This understanding of brightness serves most environments properly, as it caters to each channel equally. However, HSI still suffers from the inherent luminance of certain colors, and fails to compensate for them.

HSL

HSL, a brightness A. Leone and C. Distante. Shadow detection for moving objects based on texture analysis. Pattern Recognition, 40(4):12221233, 2007. representation called *Lightness*, is called a 'bi-hexcone' model [ref bihex]

Lightness is an average of the primary and tertiary color components:

$$L = \frac{\max(R, G, B) + \min(R, G, B)}{2} \quad (1.7)$$

Excluding the secondary color component has the effect of translating the brightness plane described by HSI. While perceptually similar to other brightness models, HSL provides more balanced values when one channel's value is in extrema.

Relative Luminance (Y)

Originally issued in 1982, Rec. 601 [ref?] defines one of the first standards for converting analog signal into digital video. *Relative Luminance (Y)*, or, when gamma-corrected, *Luma (Y')* is the simplest extrapolation of perceptually-relevant brightness. Luminance is defined as a coefficient-weighted linear combination:

$$Y = 0.299R + 0.587G + 0.114B \quad (1.8)$$

Rec. 709 later modified the coefficients to 0.21, 0.72, and 0.07, respectively. For this study's experimental purposes, Rec. 601 coefficients were used. This brightness model highlights the human eye's sensitivity to green hues, and is therefore particularly relevant when observing outdoor scenes.

Euclidean Norm

Taking the Euclidean norm of an $R\vec{G}B$ vector is measuring the three-dimensional distance from absolute black, $R\vec{G}B = (0, 0, 0)$.

$$Norm = \sqrt{\Delta R^2 + \Delta G^2 + \Delta B^2} \quad (1.9)$$

While the Euclidean norm does not weight the channels perceptually, as Luminance does, it does represent the most accurate way to determine the color difference between two $R\vec{G}B$ vectors, and therefore proves valuable when comparing foreground pixels to background pixels.

HSP

HSP, or *Perceived Brightness*, combines the three dimensional distance of Euclidean norm, and the perceptually weighted coefficients of Luminance. HSP was introduced by Darel Rex Finley in 2006.

$$P = \sqrt{0.299\Delta R^2 + 0.587\Delta G^2 + 0.114\Delta B^2} \quad (1.10)$$

Ideally, HSP provides the most accurate perceptually conscious brightness. Environments that experience large saturation shifts due to shadows benefit primarily, as both color distance and weighted coefficients are considered.

1.2.5 Low-contrast SIFT Keypoints

A scene typically contains a large set of low-level textural features unique to it. These low-level features have been characterized and quantified in many ways, such as the SIFT, SURF, or FAST feature descriptors [??, ??, ??]. These feature descriptors are used for image recognition and retrieval at various scales, rotations, and translations. These algorithms are traditionally performed on intensity images, i.e., grayscale images. However, hue and saturation play a large role in scene characterization; therefore, SIFT implementations were developed to incorporate color information. Popular implementations of a color-SIFT method are HSV-SIFT, RGB-SIFT, Hue-SIFT, and others [??, ??, ??].

Intensity images, while discarding chromatic information, often retain structural and textural information, due to image gradients' invariant basis in intensity. We assume a cast shadow similarly has minimal impact on underlying structures and textures, due to success of Textural shadow removal methods [??]. As a result, SIFT keypoints remain largely invariant between a frame and its background model in intensity images. Since cast shadows do not modify the underlying textural structure, they do not affect the detection of SIFT keypoints.

In a traditional understanding of shadow attenuation as a linear process, both the hue and saturation of a pixel remain constant, as the intensity attenuates in a shadow region. Since this study assumes a different, non-linear understanding of light attenuation, we sought for representative changes that

shadows bring to the hue and saturation channels of an HSV image. By detecting and displaying SIFT keypoints in the saturation channel alone, we observe qualitatively small localized structure changes within these shadowed regions (Figure ??).

We attempt to characterize these localized changes in structure using HSV-SIFT, by detecting SIFT keypoints using only the saturation channels of both a frame and its corresponding background model. It is apparent that foreground objects introduce significant structural changes in any of the three channels, therefore the collection of SIFT descriptors is limited to only those considered *low contrast*. The SIFT algorithm operates in four major stages: scale-space extrema extraction, elimination of low-contrast keypoints, elimination of strong edge responses, and the assignment of orientations. Low-contrast detections are eliminated due to their susceptibility to noise [??]. In our study, we build two matrices: the first containing keypoints of low and normal contrast (still excluding edge responses), and the second containing only normal contrast keypoints. We then extract low-contrast keypoints using an exclusive-or operation. The non-linearity of light attenuation, paired with previously observed structural changes, make shadowed regions in the saturation channel ripe regions for low-contrast SIFT keypoints. When drawn onto the source frame, we can see that low-contrast SIFT keypoints align logically with shadowed regions (Figure ??).

With shadows partially characterized by low-contrast SIFT keypoints, this study utilizes this insight by modeling the approximate proportion of



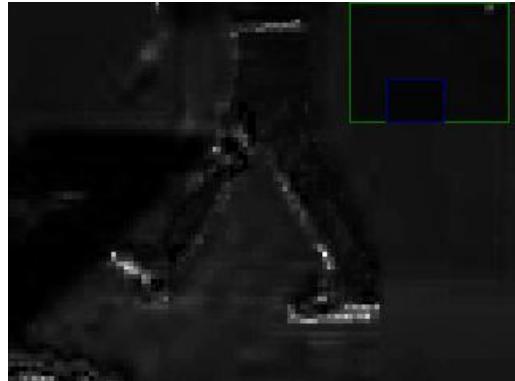
(a) aton_lab



(b) aton_lab ground truth



(c) Saturation channel (background)



(d) Saturation channel (frame)

Figure 1.18: The limited structural effects of cast shadows in the saturation channel (d).



(a) Low-contrast SIFT keypoints detected using the intensity channel.



(b) Low-contrast SIFT keypoints detected in the saturation channel.

Figure 1.19: Detecting low-contrast SIFT keypoints in the saturation channel more effectively captures structural changes due to cast shadows.

shadows introduced in a given frame. The quantity of detected SIFT features in a frame f is represented as $SIFT_c(f)$, where c represents the contrast ratio parameter used. The ratio of total low contrast SIFT features to normal SIFT features is calculated for a single frame in Eqn. ??.

$$SIFT_{\%C}(f) = 1 - \frac{SIFT_{0.04}(f)}{SIFT_{0.01}(f)} \quad (1.11)$$

$SIFT_{0.04}$ represents features captured using the default contrast threshold (0.04), and $SIFT_{0.01}$ represents features captured using a lower contrast threshold. We then compare the $SIFT_{\%C}(f)$ ratios of the foreground and background frames. A ratio is found between the two in Eqn. ??.

$$SIFT_{fg/bg} = \frac{SIFT_{\%C}(fg)}{SIFT_{\%C}(bg)} \quad (1.12)$$

The resultant $SIFT_{fg/bg}$ estimates the percentage change of the number of detected low contrast SIFT features from the background to the foreground frame. We assume a greater presence of low contrast SIFT features loosely indicates a greater quantity of shadows in a frame. In order to boost correlation between a parameter and a primary environmental property, $SIFT_{fg/bg}$ is multiplied against the value of an observed environmental property, serving as a scaling factor. We use this scaling factor to perform sensitivity analysis using the parameter $coneR1^*$ and the environmental property α . Results of this analysis are found in section ??.

1.3 Weak Detector Estimation - Creating a Model

With $coneR1^*$ calculated for a wide variety of datasets, we can begin to model a difference between environmental properties and the optimal parameter. While the attenuation from the foreground to the background displays a correlative relationship to $coneR1^*$, this correlation does not indicate a relationship to the magnitude of $coneR1^*$. This is due to the nature of the properties of attenuation. The attenuation calculation utilized is a function of brightness shift as a percentage of a background value, e.g., a pair of foreground/background pixels with a value of 50 and 100 produces the same attenuation as a pair valued at 25 and 50, respectively. Normalizing correlative environmental properties, we model an optimal shift from observed properties to the optimal value of the weak detector, i.e., the shift is modeled as $shift_{req.} = (coneR1^* - \alpha)$.

$shift_{req.}$ can be extrapolated from the magnitude of color shift (ΔRGB) found from foreground to background. By plotting ΔRGB against $shift_{req.}$, we produce a general relationship. Using data points from each frame in each dataset, we can produce a best-fit polynomial to generalize a model of attenuation and brightness magnitude shift into a function dictating the required shift of attenuation (shown in Figure ??). This required shift is defined as $shift_\alpha$. To avoid over-fitting, a parametrized logarithm was chosen to represent the required magnitude shift (ΔRGB). The logarithm used to calculate

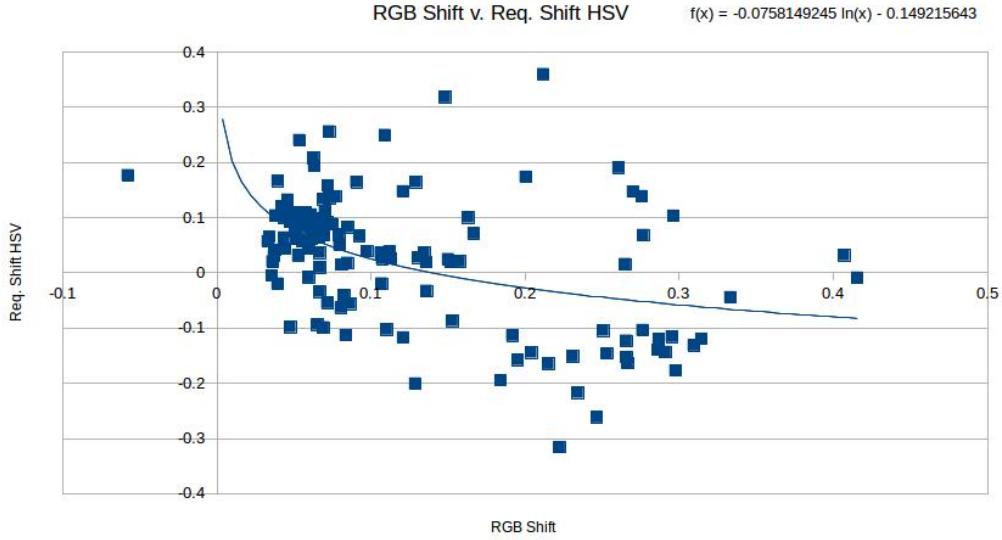


Figure 1.20: Regression performed on RGB shift found per frame, vs. required magnitude shift to optimal.

$shift_\alpha$, derived from Figure ??, is defined in Eqn. ??.

$$shift_\alpha = -0.11307 * \ln(\Delta RGB) - 0.1884 \quad (1.13)$$

We are now able to calculate a new algorithmic parameter, $coneR1'$.

$coneR1'$ is calculated per frame, per environmental properties. Calculating

$coneR1'$ is shown in Eqn. ??.

$$coneR1' = (1 - \Delta RGB) * (1 - \alpha_{\% \Delta}) + shift_\alpha \quad (1.14)$$