ERSP Proposal: Accelerating Shadow Detection and Removal Methods

Alvin Wang, Edward Ding, Kyle Zhao, Jennifer Zhu,
Advisors: Professor Beyeler, Galen Pogoncheff, Apurv Varshney

December 2023

1 Context and Motivation

1.1 Motivation

The existence of shadows in images can drastically affect interpretation and accuracy across a variety of different computer vision tasks, such as computational photography and augmented reality. Consequently, the efficient identification and management of shadows is a necessary operation. While existing shadow detection/removal algorithms produce accurate and realistic shadow-free images, most existing implementations are not optimized to be either memory or runtime efficient. For example, the shadow detection/removal algorithm presented in [1] takes upwards of minutes to process a relatively simple 500 x 500 pixel image. Additionally, deep learning-based models rely on expensive, state-of-the-art hardware to achieve runtimes that are still unsatisfactory.

For usage in the most latency-critical applications, such as autonomous driving, augmented reality/virtual reality and robotics, substantial efficiency improvements are necessary to achieve viability. A multi-minute runtime per frame for a given algorithm or implementation limits its usefulness to static image datasets. [16] notes that in cases of autonomous driving, for example, "every millisecond that elapses during object recognition is a millisecond not available for effecting safe operation." In this case, minimizing processing latency has a direct positive impact on the safety of passengers and passerby and improves the feasibility of autonomous driving as a whole.

Our project aims to remedy these shortcomings by benchmarking and optimizing the accuracy, runtime and memory efficiency of existing models, with a focus on unlocking real-time video processing for shadow detection and removal. Most importantly, these improvements must be made without substantially compromising the accuracy of the model or algorithm.

1.2 Context

1.2.1 The Problematic Phenomenon That is Shadows

The purpose of technology is to meet the needs of human beings, which is why there is significant contemporary emphasis on areas that aid or even entirely replace human input, such as autonomous robotics, augmented reality, and automatic surveillance systems. Often, the trivial solution to these problems is to adapt technology to understand and perceive the world the same way humans do. However, in doing so, this can result in problems that may be unintuitive to humans. As an example, a seemingly unintuitive error that affects the performance of computer vision technologies is the presence of shadows within images. Since shadows cause regions of relative darkness within images, this resultant darkness is directly translated to the individual pixel data of the image. These dark regions can cause an algorithm to misidentify shadows as objects or misrepresent the properties and dimensions of objects, resulting decreased accuracy of object detection and recognition algorithms [10]. The current preferred solution to this problem is to pre-process the image, removing shadows entirely from the image before it is sent to the rest of the pipeline [7].

1.2.2 'Classical' Computer Vision Approach

The earliest attempts at shadow detection and removal are commonly referred to as 'Classical' computer vision techniques. 'Classical' computer vision techniques rigorously process and define an image using mathematical equations and systems, then attempt to "undo" the shadow with an emphasis on preserving realism [1].

1.2.3 Physics Behind Shadows and 'Classical' CV cont.

One foundational 'Classical' shadow detection paper published in 2011 presented the idea of approximating regions of shadows in a 500 x 500 pixel image using surfaces [1]. According to this approach, shadows are produced when an object obscures a light source, producing a nonuniform shadow that is separated into the umbra (darkest region of the shadow), penumbra regions, which vary in illumination (brightness) intensity, and non-shadowed regions.

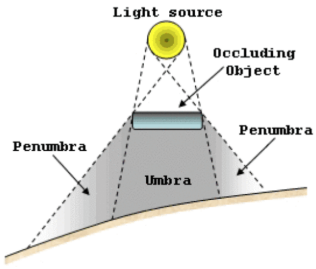

Figure 1: 2-D Example of a shadow produced from an object obscuring a point light source [1].

The obscuring of a light source in images typically creates a region of reduced illumination intensity (shadow), which can be modeled through physics formulas and represented by a 3-D surface. This representation is significant because illumination intensity surfaces allow the shadow removal algorithm to recognize global shadow and illumination features, preserve original image texture and quality, and eliminate unwanted image artifacts (errors). However, 'Classical' approaches tend to require rigorous mathematical experimentation and the manual identification of shadow boundaries to form accurate shadow masks, a black-and-white binary version of the original image that identifies shadowed regions [15]. Consequently, researchers sought more automatic processes to improve efficiency, scalability, and consistency.

1.2.4 Deep Learning Computer Vision Approach

Beginning around 2012, researchers began applying computer vision deep learning frameworks to the problem of shadow detection/removal [6, 8, 5, 18]. Deep learning models function with an end-to-end approach, meaning that a single trained model can handle an entire task, such as shadow detection, automatically. End to end models do not require researchers to manually calculate rigorous mathematical formulas to analyze image features, which greatly reduces human intervention.

[6], published in 2014, presents a novel deep learning neural network for the task of shadow detection and generation of shadow masks. Neural networks, the fundamental building blocks of deep learning, are based on the human brain, a highly complex, interconnected network of electric signals and neurons that process information. A neural network consists of an input layer, which receives raw data, a series of hidden layers to capture features from the data, and an output layer, which outputs final predictions. The paper presents a framework that uses a series of convolutional layers, which are deep learning layers used in image-related or computer vision tasks, to capture global shadow features from images. In [6], these features are combined and processed to produce the final output: a shadow mask of the original image.

Figure 2: Images on the left are images with shadows, while images on the right are the generated shadow masks of their left counterparts. Shadow masks are black-white binary versions of images that signify the location of shadows [18].

1.2.5 Generative Adversarial Networks

However, papers that presented deep learning shadow removal methods would only be published years later. Around 2017, researchers began publishing papers on shadow detection and shadow removal that involve the use of Generative Adversarial Networks (GANs), a type of machine learning algorithm that pits a generative model against an adversary or discriminator. These GANs recognize that shadow detection and shadow removal are intimately related tasks and are operated in an end-to-end fashion. After being trained on a large dataset, GANs are typically able to automatically produce a high-quality, shadow-free version of an input image.

GANs consist of a generator and a discriminator, which each consist of a series of deep learning layers. The goal of the generator is to take in an input, usually random noise drawn from a probabilistic distribution, and produce an output that is as close as possible to real data from a training set. The goal of the discriminator is to take the generated data from the generator and distinguish if it was generated or sourced from the real data set. Both the generator and discriminator are trained in a "zero-sum game." If the discriminator determines a generated image is real, the discriminator is punished and the generator is rewarded. Likewise, if the discriminator determines a generated image is fake, the discriminator is rewarded and the generator is punished. As the GAN is trained on more data, the generator becomes better at generating images indistinguishable from those in the real data set and the discriminator becomes better at identifying generated images. In the context of shadow removal and shadow detection, given an input of a large data set of shadowed and shadow-free images, the discriminator can be thrown out and the generator becomes accurate at producing shadow-free images from shadowed images. In regards to accuracy and generalizability, state-of-the-art shadow detection and removal methods achieve significant and promising results, as shown in Figure 3.

1.2.6 Optimizing Deep Learning Methods

However, there is a gap in the literature in regards to efficiency and time-complexity of said methods, which is what our research group aims to resolve.

1.3 Literature Review

1.3.1 'Classical' Works

Current state-of-the-art deep learning shadow detection and removal frameworks are built off of numerous "classical" computer vision frameworks. Foundational shadow detection methods include the approximating of shadows as surfaces and creating inferences of their respective light sources based on that information [1],

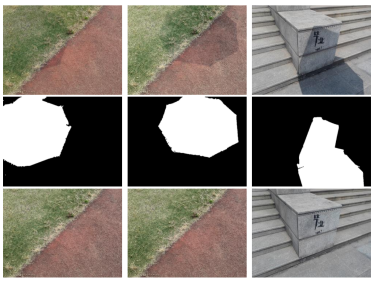

Figure 3: The top row represents three original images with shadows. The middle row represents the shadow masks of their respective images. The bottom row represents generated, shadow-free versions of the original images [11].

detecting the edges of shadows using retinex algorithms (analyzing the consistency of color in input images) [3] and conversion of image pixel data into a mathematical equation [14].

Foundational shadow removal methods include taking advantage of color lines, a pixel's value in a RGB (color) space that remains constant even if there is a variation in illumination intensity, to determine and undo shadows [17] and pixelwise multiplication of shadowed pixels with calculated shadow factors, which are mathematical representations of shadows used to increase illumination of shadowed areas [1].

1.3.2 Deep Learning Works

In regards to deep learning, many state-of-the-art models utilize Generative Adversarial Networks to generate shadow-free images [5, 2, 12, 18, 11]. However, one key shortcoming of many of the aforementioned deep learning models is data scarcity and a lack of generalizability to other datasets [18]. To combat this issue, a recent study used an image rendering engine (Unity Game Engine) to produce synthetic tripled data to feed into a GAN [5]. [2, 12, 18, 11] all use supervised learning (a subset of deep learning where models are trained on labeled data) frameworks, which necessitate datasets that consist of original, shadowed images, their shadow-free counterparts, and shadow masks. These datasets are called tripled datasets. Even though synthetic datasets can significantly increase the amount of available data to train models, leading to more accurate models, they tend to run into the issue of domain bias. To address domain bias, which occurs when a model performs well on its training data but poorly on untrained data, real-world data was converted into a synthetic style using style transfer [5]. Style transfer, which applies one image's style to another's content, enhances the shadow detection algorithm's effectiveness with diverse data.

Additionally, instead of using an end-to-end framework like [2, 12, 18, 11], the shadow detection and shadow removal models of [5] were trained separately in recognition of the potential issue of domain bias.

1.3.3 Optimization of Shadow Detection/Removal Works

In regards to optimization of shadow detection and shadow removal, recent substantial advancements in hardware computing power have the potential to greatly improve the performance of a large subset of shadow removal algorithms. One paper discusses the application of Graphical Processing Units (GPUs) to accelerate the capabilities of 'Classical' approaches to shadow detection and shadow removal by separating the algorithm into distinct, independent tasks and processing all kernels in parallel. While the algorithm achieved a 21.67x speedup for shadow removal on images with specific conditions, the paper failed to generalize the algorithm for all cases of shadow removal and detection, as the authors manually re-implemented an existing MATLAB based implementation in custom GPU-optimized CUDA-based code [9]. This approach, while yielding impressive improvements, is not scalable due to its highly customized nature. Additionally, this solution is only useful for certain unoptimized implementations of classical-CV-based algorithms, not deep learning-based models. More work must be performed to find a more general solution.

1.3.4 Optimization of Deep Learning Inference

A method to increasing the runtime and memory efficiency of deep learning inference is to use integer quantization, which compresses the information in a model by reducing the precision of the weights or activations of the deep neural network. Integer quantization aims to speed up inference by using less memory and having faster mathematical computation times, allowing use of neural networks on less powerful computation devices, including edge devices [4]. However, integer quantization does have drawbacks. Due to loss of variable precision, quantizing a neural network can lead to loss of accuracy.

[4] observes that integer quantization from floating point 16 (fp16) to 8-bit integer (int8) data types can bring upwards of a 3.00x speedup in inference performance for a given model. [13] analyzes effects of integer quantization on accuracy and proposes a workflow involving Partial Quantization and Quantization-Aware Training. Partial Quantization involves testing the sensitivities of layers and using Quantization-Aware training, a method that actively quantizes the neural network's weights and activations while training, to maintain more than 99.9% accuracy of the original models' accuracy.

2 Proposed Solution

We propose a novel solution to reduce latency by removing redundant convolutional layers and applying integer quantization without significant loss of accuracy.

2.1 Layer Optimization

State of the art supervised learning models capture a hierarchical representation of image features using series of convolutional layers. These layers each take in an image tensor as an input, a three-dimensional array that typically consists of height, width, and image channels. Learnable filters or kernels are then slid over input tensors, performing pixelwise multiplication to capture specific features from tensors, which produces feature maps.

Network	Layer	Cv_0	Cv_1	Cv_2	Cv_3	Cv_4 (×3)	Cv_5	CvT_6	CvT_7 (×3)	CvT_8	CvT_9	CvT_{10}	CvT_{11}
G_1/G_2	#C_in	3/4	64	128	256	512	512	512	1024	1024	512	256	128
	#C_out	64	128	256	512	512	512	512	512	256	128	64	1/3
	before	–	LReLU	LReLU	LReLU	LReLU	LReLU	ReLU	ReLU	ReLU	ReLU	ReLU	ReLU
	after	–	BN	BN	BN	BN	–	BN	BN	BN	BN	BN	Tanh
	link	$\rightarrow CvT_{11}$	$\rightarrow CvT_{10}$	$\rightarrow CvT_9$	$\rightarrow CvT_8$	$\rightarrow CvT_7$	–	–	$Cv_4 \rightarrow$	$Cv_3 \rightarrow$	$Cv_2 \rightarrow$	$Cv_1 \rightarrow$	$Cv_0 \rightarrow$

Figure 4: Example of the deep learning architecture for the generator of a GAN; Table represents a series of convolutional layers, which each capture specific features of an image. "C_in" and "C_out" denote the amount of input channels and output channels respectively.

In our project, we plan to utilize PyTorch to conduct layer ablation studies, a crucial process for optimizing neural networks in shadow detection and removal. This technique involves initially training the model, then selectively removing or modifying certain layers to evaluate their impact. Referring to Figure 4, we could remove pairs of related convolutional layers, such as Cv_0 and CvT_{11}, in the generator architecture and test for loss in accuracy. Through this process, we aim to discern the importance and efficiency of each layer by observing changes in accuracy, feature capture, and computational cost. Post-ablation, the model is fine-tuned to recover accuracy, and its accuracy (both pre and post-ablation) is rigorously benchmarked. This approach enables us to identify and eliminate redundant layers, resulting in a more efficient and effective neural network tailored to the specific demands of shadow detection and removal.

2.2 Integer Quantization

Additionally, we propose the use of integer quantization to make models more memory efficient and faster to execute. Integer quantization is the reduction of precision of a model's parameters from floating-point numbers to integers. Floating-point numbers are typically represented by 16 or 32-bits and can be mapped

to integers of reduced bit width, typically 8 bits with varying loss of accuracy, which conserves memory. These parameters are converted into integers typically through multiplication by scale factors, which are learned.

We plan on applying two types of integer quantization methods to a range of deep learning models: affine quantization, which works on quantizing a range of values that are not symmetrical around 0, and scale quantization, which works on quantizing a symmetrical range $[-\alpha, \alpha]$. Due to having a range not centered at 0, affine quantization is less efficient but also provides more flexibility, preserving more data on skewed data.

We plan on applying integer quantization to the weights and activations of the aforementioned convolution layers, the loss functions of the generator and discriminator of the Conditional Generative Adversarial Networks (GAN), activation functions, such as Leaky Rectified Linear Units, and batch normalization parameters. We would start by training the neural network on the given dataset with full precision (32-bit floating point numbers). Then, we would quantize the weights and activations by manually converting floating-point parameters into integers. Finally, we would fine-tune the quantized model on the original dataset with the aim of recovering loss of accuracy. To further reduce the loss of accuracy caused by integer quantization, a quantization-aware training process can be applied[13]. In quantization-aware training, scaling factors are treated as learnable parameters and are updating during the training process to minimize loss. This newly quantized model would also be benchmarked using processes outlined below.

3 Evaluation and Implementation Plan

To evaluate the performance of our modified shadow removal algorithm, we will benchmark both the efficiency and accuracy of our modified models as well as existing baseline models on the publicly available dataset called ISTD which contains 1,870 triplets of a shadowed image, a shadow mask and a shadow-free image [12]. The images have been taken under a variety of environments. Similarly to the dataset used by Wang et al. in [12], we will randomly select 1,330 triplets for training the models and use the remaining 540 for testing.

3.1 Evaluating Accuracy

To measure the accuracy of our modified removal model, we adopt the accuracy evaluation plan used by Qu et al. in [8] by using the root mean square error (RMSE) and the structure similarity index (SSIM) to compare the resulting shadowless image and the original shadow free image. RMSE calculates the error of each pixel of the removal model's output with the original non-shadow image. A lower RMSE indicates that the images are more similar. SSIM measures the contrast and luminance of the overall image as well as the difference in structural information between the model's output and the original non-shadow image. The SSIM value is between -1 and 1 with a SSIM of 1 indicating that the images are identical.

3.2 Evaluating Efficiency

To measure the efficiency of our modified removal model, we will calculate the average inference time for the images in the test set. The level of memory utilization during training and inference will also be recorded. All of these tests will be run on a reference machine on the same set of images.

3.3 Evaluating Success

Success in our project is determined by achieving measurable improvements in neural network efficiency, guided by benchmarks observed in related research. While existing literature suggests enhancements around 20%, we aim for any progress in this direction, no matter how modest. Our specific goals include reducing memory usage and decreasing latency. A significant reduction in process time, from the current one minute to a matter of seconds, alongside enhanced memory efficiency, would mark a successful implementation of

our proposed modifications and signify a meaningful step forward in expanding the practical applications of our neural network model.

3.4 Implementation Plan

Fall 2023: Gain more understanding on the topic and finalize on the proposal

- Weeks 10 - Finalize the project proposal and final presentation.

Winter 2024: Finish up learning and start on our project

- Weeks 1-2
  - Finish set up Keras/Pytorch software
  - Learn basic machine learning through online coursework
  - Understand paper from Galen
- Weeks 3-4
  - Select images for dataset
  - Experiment with [11] code
- Week 5
  - Initial Benchmarking for given dataset for existing model
- Weeks 6-7
  - Runtime/efficiency analysis for existing model
- Weeks 7-8
  - Edit existing model to incorporate integer quantization
- Week 9
  - Benchmark new model after implementation of integer quantization
- Week 10
  - Begin implementation of new memory optimization methods

Spring 2024: Finish up project, evaluate our results, presentation

- Weeks 1-2
  - Finish optimization for memory access for new model
- Week 3
  - Benchmark new memory-optimized model
- Week 4
  - Incorporate a second round of memory access optimizations into the model
- Week 5
  - Re-benchmark model on dataset
- Weeks 6-10
  - Collect results, make project posters, and prepare for presentation

References

[1] Eli Arbel and Hagit Hel-Or. Shadow removal using intensity surfaces and texture anchor points. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(6):1202–1216, 2011.

[2] Bin Ding, Chengjiang Long, Ling Zhang, and Chunxia Xiao. ARGAN: attentive recurrent generative adversarial network for shadow detection and removal. *CoRR*, abs/1908.01323, 2019.

[3] Graham D Finlayson, Steven D Hordley, and Mark S Drew. Removing shadows from images using retinex. In *Color and imaging conference*, volume 2002, pages 73–79. Society for Imaging Science and Technology, 2002.

[4] Amir Gholami, Sehoon Kim, Zhen Dong, Zhewei Yao, Michael W. Mahoney, and Kurt Keutzer. A survey of quantization methods for efficient neural network inference. 2021.

[5] Rui Guo, Babajide Ayinde, and Hao Sun. Efficient shadow detection and removal using synthetic data with domain adaptation. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 5867–5874, 2021.

[6] Salman Hameed Khan, Mohammed Bennamoun, Ferdous Sohel, and Roberto Togneri. Automatic feature learning for robust shadow detection. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1939–1946, 2014.

[7] Chin-Teng Lin, Chien-Ting Yang, Yu-Wen Shou, and Tzu-Kuei Shen. An efficient and robust moving shadow removal algorithm and its applications in its. *EURASIP Journal on Advances in Signal Processing*, 2010:1–19, 2010.

[8] Liangqiong Qu, Jiandong Tian, Shengfeng He, Yandong Tang, and Rynson W. H. Lau. Deshadownet: A multi-context embedding deep network for shadow removal. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[9] Edward Richter, Ryan Raettig, Joshua Mack, Spencer Valancius, Burak Unal, and Ali Akoglu. Accelerated shadow detection and removal method. In *2019 IEEE/ACS 16th International Conference on Computer Systems and Applications (AICCSA)*, pages 1–8, 2019.

[10] J. Stander, R. Mech, and J. Ostermann. Detection of moving cast shadows for object segmentation. *IEEE Transactions on Multimedia*, 1(1):65–76, 1999.

[11] Jifeng Wang, Xiang Li, and Jian Yang. Stacked conditional generative adversarial networks for jointly learning shadow detection and shadow removal. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1788–1797, 2018.

[12] Jifeng Wang, Xiang Li, and Jian Yang. Stacked conditional generative adversarial networks for jointly learning shadow detection and shadow removal. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[13] Hao Wu, Patrick Judd, Xiaojie Zhang, Mikhail Isaev, and Paulius Micikevicius. Integer quantization for deep learning inference: Principles and empirical evaluation. 2020.

[14] Tai-Pang Wu and Chi-Keung Tang. A bayesian approach for shadow extraction from a single image. In *Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1*, volume 1, pages 480–487 Vol. 1, 2005.

[15] Tai-Pang Wu, Chi-Keung Tang, Michael S Brown, and Heung-Yeung Shum. Natural shadow matting. *ACM Transactions on Graphics (TOG)*, 26(2):8–es, 2007.

[16] Ming Yang, Shige Wang, Joshua Bakita, Thanh Vu, F. Donelson Smith, James H. Anderson, and Jan-Michael Frahm. Re-thinking cnn frameworks for time-sensitive autonomous-driving applications: Addressing an industrial challenge. In *2019 IEEE Real-Time and Embedded Technology and Applications Symposium (RTAS)*, pages 305–317, 2019.

[17] Xiaoming Yu, Ge Li, Zhenqiang Ying, and Xiaoqiang Guo. A new shadow removal method using color-lines. In Michael Felsberg, Anders Heyden, and Norbert Krüger, editors, *Computer Analysis of Images and Patterns*, pages 307–319, Cham, 2017. Springer International Publishing.

[18] Yide Zhang, Zhihong Li, Zilong Sun, Guoliang Liu, and Yichao Cao. Srodnet: Pavement crack detection based on deep convolutional neural network and shadow removal. In *2022 12th International Conference on CYBER Technology in Automation, Control, and Intelligent Systems (CYBER)*, pages 504–508, 2022.