# Locally Adaptive Thresholding for Single-Shot Structured Light Patterns

Kasra Sadatsharifi — University of Waterloo, Waterloo, ON, Canada
Mohamed A. Naiel — University of Waterloo, Waterloo, ON, Canada
Mark Lamm — Christie Digital Systems Inc., Kitchener, ON, Canada
Paul Fieguth — University of Waterloo, Waterloo, ON, Canada
Email: {ksadatsh, mohamed.naiel, pfieguth}@uwaterloo.ca, mark.lamm@christiedigital.com

## Abstract

Image thresholding is a challenging task due to its sensitivity to environmental variations and degradation in the quality of the captured image. Although many image thresholding methods have been introduced, most of them require the fine tuning of a thresholding parameter that is not suitable for single-shot structured light (SSSL) based projector-camera applications. In this paper, we introduce a locally adaptive thresholding method with automatic parameter selection based on the local statistics of the distinct image partitions. For assessing the proposed scheme, we introduce an evaluation that relies on mapping SSSL patterns between the camera and projector spaces. Experimental results demonstrate the effectiveness of the proposed technique by maintaining the thresholding accuracy of the baseline method, without the need to fine tune the obtained thresholding parameter or any noticeable change in the qualitative results.

## 1 Introduction

As a fundamental and challenging problem in the field of computer vision, 3D reconstruction has been an active area of research for decades [1], and projector-camera systems are one means to achieve this goal. The key step in creating a 3D reconstructed point-cloud for a scene is to have enough information to do so, where the most fundamental information for inferring scene geometry is a set of pixel correspondences. For projector-camera systems the projector is able to influence the scene via structured light (SL), for which there are two well known SL approaches, namely multi-shot SL [2], and single-shot SL [3]. Although the multi-shot SL generates dense pixel correspondances, they require multiple pattern to be projected over time, which limits their applicability to a static configuration setup. In contrast, single-shot SL approach lifts this limitation by only requiring a single scene capture, but at a cost of sparser pixel correspondences. In [3], the single-shot structured light (SSSL) approach captures and binarizes the projected pattern, hence image binarization will impact the quality of the whole SSSL process.

There is a significant literature on image thresholding, including in the SSSL context, and both global [4–6] and local [7–9] image thresholding methods have been used [10, 11]. In global thresholding [4, 5] a single constant is used to threshold every pixel in the image, causing the method to suffer from a susceptibility to spatial intensity variations. On the other hand, local image thresholding-based methods [7–9] are more robust to intensity variations, since they utilize a locally determined threshold for every pixel in the image. However, these methods [8, 9] usually need a parameter fine-tuning step to accommodate environmental variations, which is either time consuming or itself subject to error.

In this paper, we introduce a locally adaptive thresholding method that partitions a camera image into distinct blocks and extracts local statistics from each block. Next, we show that every pixel in the image can be thresholded based on the local average multiplied by a hyper parameter that can be estimated based on a novel closed form expression. The proposed technique leverages a block smoothing process for the thresholds to avoid the blocking artifacts that may be caused due to block partitioning the image. A new evaluation strategy is introduced for the first time, to the best of our knowledge, which maps the thresholded camera pixels into the projector space and therefore the binary SSSL patterns are considered as the ground truth. Quantitative and qualitative experimental results demonstrate the effectiveness of the proposed thresholding scheme compared to the baseline that requires parameter fine tuning.
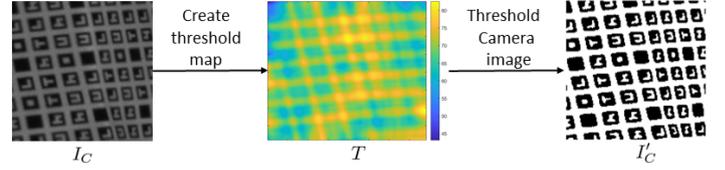


*Fig. 1:* Block diagram of the local thresholding method, where $I_C$ and $I_C'$ are the input and thresholded camera image, and $T$ is the thresholding map for every pixel. Neighborhood size of the smoothing filter is $3 \times 3$.

## 2 Background

In the literature, there exist various image thresholding methods that have been proposed [4, 5, 8, 9]. For instance, Otsu's method [4], which is a global thresholding method, aimed to maximize the between class variance in order to set a global threshold. However, it cannot be used in projector-camera systems SSSL applications, due to the varying illumination of the scene and the distortion caused by the shape of the object. Hence, the adaptive thresholding methods in which a local threshold is set for each pixel (so called locally adaptive thresholding) are of interest to the target application, such as the methods in [8, 9]. Reader would also be referred to [12, 13] for more information on different types of related image thresholding methods.

**Sauvola and Pietikäinen's Method:** In this section, a brief overview on Sauvola and Pietikäinen (SP) method in [8] is presented, which is originally developed for document image thresholding. Given an input camera image $I_C$, this method uses the local mean $m(x,y)$ and the local standard deviation $\delta(x,y)$ computed at the $(x,y)$th pixel location to set a value, $T(x,y)$, for thresholding the $(x,y)$th camera pixel. More specifically, the following relationship is used to obtain $T(x,y)$:

$$T(x,y) = m(x,y)\left(1 - k\left(1 - \frac{\delta(x,y)}{R}\right)\right) \quad (1)$$

where $R$ is the dynamic range of the image pixels, which is a way to standardize the pixel intensities in the image, and $k$ is the bias parameter. An illustration of SP's thresholding method on a sample image is shown in Fig. 1. It is noted that in the context of projector-camera systems, the parameter $k$ changes with different surfaces and different illuminations. Therefore, fine tuning this parameter is very time consuming and is not possible to do during the SSSL process.

## 3 Proposed Method

Unlike SP's method that requires fine tuning for the parameter $k$ in (1) for every image, our proposed method aims to obtain automatically an optimal $k$ in order to adaptively threshold each image pixel within a local region. Fig. 2 shows the overall architecture of the proposed method. In this method, $I_C$ is firstly smoothed by using an average filter that leverages the integral image technique [9], which increases the speed of the filtering process. Afterwards, the smoothed image is partitioned into a number of distinct blocks, with relative size of three times the cells in the projector pattern, $I_P$, and based on the average value of the $10^{th}$ and $90^{th}$ percentiles of each block, the algorithm assigns a specific $k$ to each block as follows:

$$k(u,v) = \frac{\Gamma(u,v) - M(u,v)}{4R} \quad (2)$$

where $R$ is the image dynamic range, $k(u,v)$ is the adjusted bias value for the $(u,v)$th block. Based on the majority of white or black
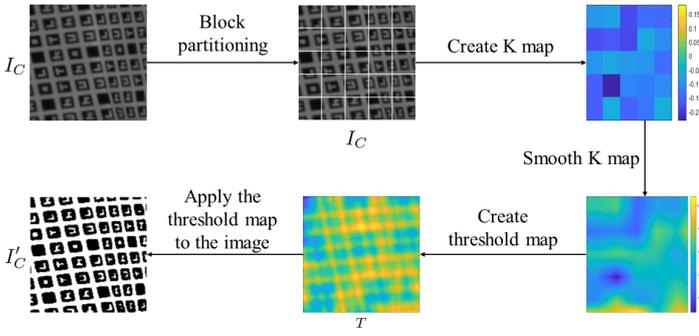
Fig. 2: Block diagram of our proposed method, where $K$-map is produced, and based on $K$-map a threshold map, $T$, is achieved. Neighborhood size of the filter is the same as SP and the block size are three times the cell sizes in the projector pattern, $I_P$



Fig. 3: Block diagram of the evaluation process, where $I_C$, $I'_C$, and $I'_{CP}$ are the captured camera image, the thresholded camera image, and the mapped thresholded camera image that is used to compare with the pattern $I_P$ and evaluate the accuracy, respectively.

pixels in each block we wish to set the threshold above or below $M(u,v)$. However, in order to prevent falling far from $M(u,v)$ and have worse threshold set for the block, we need to slightly reduce the range of variation for $k(u,v)$. Therefore, we divided the nominator of (2) by the constant $4R$. Additionally, $\Gamma(u,v)$ is defined as

$$\Gamma(u,v) = \frac{p_{10^{th}}(u,v) + p_{90^{th}}(u,v)}{2} \tag{3}$$

where $M(u,v)$ is the mean of the $(u,v)$th block, and $p_{10^{th}}$ and $p_{90^{th}}$ are the $10^{th}$ and $90^{th}$ percentiles of the pixel intensities within the same block, respectively.

After defining $k$ for each block, a $K$-map that defines the thresholds per pixel is created as shown in Fig. 2. In order to avoid the blocking artifacts, a smooth version of the $K$-map is constructed using the average filter, which we denote it by $\hat{K}$. Finally, the smoothed $K$-map, i.e., $\hat{K}$, which has the values of $k$ corresponding to each pixel in the image, is then used to create the threshold map, $T$, by:

$$T(x,y) = m(x,y)\left(1 - \hat{K}(x,y)\left(1 - \frac{\delta(x,y)}{R}\right)\right) \tag{4}$$

where $m(x,y)$ and $\delta(x,y)$ are the local mean and local standard deviation of each pixel, and $\hat{K}(x,y)$ is the bias parameter by which we decide how far below or above $m(x,y)$ we need the threshold to be set. The usual values for the parameter $k$ suggested in [8, 13] and in (1) would not necessarily be the optimum value for the SSSL application. On the other hand, as can be seen in (2)-(4) our method can compute $\hat{K}(x,y)$ automatically and therefore is robust to several environmental challenges, such as shadows and distortions.

Finally, the thresholded image, $I'_C$, is achieved by applying the acquired threshold map, $T$, on the camera image, $I_C$, as follows:

$$I'_C(x,y) = \begin{cases} 1, & \text{if } I_C(x,y) \geq T(x,y) \\ 0, & \text{if } I_C(x,y) < T(x,y) \end{cases} \tag{5}$$

## 4  Experimental Results

For the experimental setup, we have used a Flea 3 camera with resolution of 2448x2048 and Christie Digital projector with resolution of 1920x1200. We tested the proposed method using SSSL patterns projected on three different 3D-printed objects named by *Wolf Head*, *Zigzag* [14] and *Building Facade* [14].

**Evaluation Methodology:** Fig. 3 shows a diagram for our proposed evaluation methodology. To evaluate the local thresholding techniques, we first generate a dense set of pixel-correspondences between the camera and projector images, $\phi_{gt}$, using a multi-shot graycode structured light patterns [15]. Our assessment strategy is based on mapping a thresholded pixel in a given camera image into the projector space leveraging the dense set of correspondences, $\phi_{gt}$, and then compare the mapped value to that in the original binary projector pattern, $I_p$. Considering the original SSSL binary pattern, $I_P$, that needs to be projected on the surface as the ground-truth image due to it not being affected by the surface geometry deformation and every pixel has a specific binary value. This approach will help us having a rich set of pixels to compare with the ground-truth.
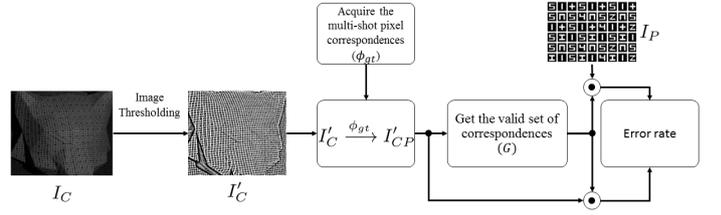
To apply this strategy, given an input SSSL camera image, $I_C$, we obtain the binarized image, $I'_C$, by applying our proposed method, or any other thresholding method in comparison. Based on the acquired pixel-correspondences between the camera and projector images, $\phi_{gt}$, we map the binarized camera image, $I'_C$, into the projector space, $I'_{CP}$. Next, for every pixel location, $(x,y)$, in $I'_C$ when all the neighboring camera pixels that are within $\eta \times \eta$ region centred around $(x,y)$, $\aleph_{x,y}$, have the same value, then we assume this correspondence is a valid pixel to compare the mapped version, $I'_{CP}$, with that in the projector pattern, $I_P$. This process can be expressed as follows:

$$G(x,y) = \begin{cases} 1, & \text{if } I'_C(x,y) = I'_C(m,n) | \forall (m,n) \in \aleph_{x,y} \\ 0, & \text{otherwise} \end{cases} \tag{6}$$

where $G(x,y)$ is the set of valid pixel-correspondences that can be used to assess $I'_{CP}(x,y)$ against $I_P(x,y)$, and the dimensions of the neighborhood $\eta \times \eta$ are set to $3 \times 3$ in this evaluation.

It is worth mentioning that checking the validity of pixels in $\phi_{gt}$ is performed before mapping $I'_C$ to $I'_{CP}$. In other words, the validity is checked in the camera space on thresholded image, $I'_C$, and then mapped to the projector space, $I'_{CP}$, for evaluation.

After obtaining the valid set of correspondences, $G$, and mapping $I'_C$ to $I'_{CP}$, in order to complete the evaluation process we need to assess whether a given pixel has been binarized correctly or not. To this end, let the sets of True and False pixel-correspondences be defined as follows:

$$\mathscr{R} = \langle (x,y) \quad | \quad I'_{CP}(x,y) = I_P(x,y) \quad , \quad G(x,y) = 1 \rangle \tag{7}$$

$$\mathscr{W} = \langle (x,y) \quad | \quad I'_{CP}(x,y) \neq I_P(x,y) \quad , \quad G(x,y) = 1 \rangle \tag{8}$$

where $\mathscr{R}$ and $\mathscr{W}$ are the sets of correctly and incorrectly thresholded pixels, respectively. We use the pixel thresholding error rate ($\xi$) that can be expressed as:

$$\xi = \frac{|\mathscr{W}|}{|\mathscr{R}| + |\mathscr{W}|} \times 100\% \tag{9}$$

where $|\cdot|$ indicates the cardinality of the set. The metric $\xi$ can infer the percentage of incorrectly thresholded pixels from the total number of valid correspondences. In the rest of this section, we utilize the proposed evaluation methodology to compare the thresholding error rate of the proposed scheme with that of SP's method [8].

**Qualitative Results:** Fig. 4 shows the different qualitative results of SP's method [8] and the proposed method. As shown in this Figure, one can observe that the proposed method is less prone to thresholding the symbols in the black squares which is helpful for next steps in SSSL process. Additionally, both methods seem to have errors on edges and discontinuities in the image which is expected. This figure also represents the results of the proposed method with the correspondences incorrectly thresholded and marked as red crosses on three objects: *Wolf Head*, *Building Facade* and *Zigzag*.

It should be noted that the inherent material of the object in front of the setup is also affecting the final results. As can be observed from Fig. 4, the proposed method and the method in comparison generate a higher number of incorrect correspondences on the *Building Facade* than that on the *Wolf's Head*. This is due to material that *Building Facade* [14] has been built of is more reflective than *Wolf's Head*.

**Quantitative Results:** Fig. 5 shows the effect of changing the value of $k$ on the error rate, $\xi$ in (9), for SP method, and the error rate provided by the proposed method at the average estimated parameter

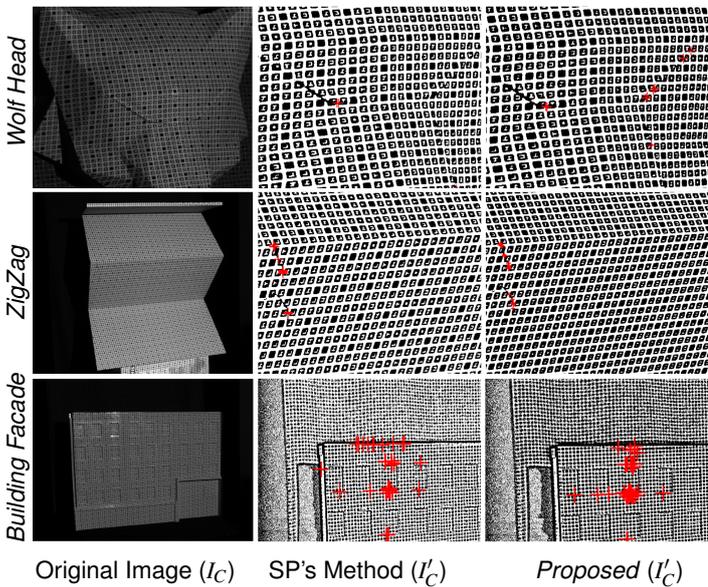| | Original Image ($I_C$) | SP's Method ($I_C'$) | *Proposed* ($I_C'$) |

**Fig. 4:** Sample qualitative results for SP's method vs our proposed method on three different surfaces, where a marker with red color denotes incorrect thresholded pixels. Note that SP's method requires adjusting $k$ to 0.2, 0.1 and 0.0 for the *Wolf Head*, *ZigZag* and *Building Facade*, respectively, for generating the best results.
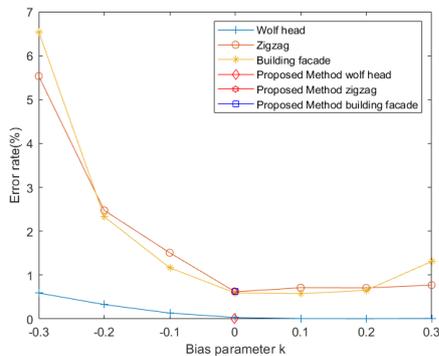


**Fig. 5:** Effect of changing $k = [-0.3, -0.2, ..., 0.3]$ on the error rate, $\xi$, for the SP's method, and the error rate provided by the proposed method at the average estimated parameter $\hat{K}$.

$\hat{K}$, where the three target surfaces are used. As shown, setting $k$ for SP's method further from the optimum value in both directions can hugely increase the error rate.

Table. 1 shows the error rate comparison between the proposed method and the SP method. As shown in the table, the error rate of the proposed method is close to that of the SP's method, which has been tuned its $k$ value to the lowest error rate in Fig. 5. Although there is not a significant difference in error between the two methods, due to automatic tuning of the hyper parameter $k$, our method is expected to offer a superior performance than the SP method with regards to our SSSL application, during which no interference from the user is needed.

## 5   Conclusion

In this paper, we have proposed a method to adaptively threshold a grayscale SSSL image using the local statistics of the image partitions. Our proposed method has the capability of automatically obtaining all the parameters needed for thresholding the image. To quantify the error rate of the thresholding methods, we have introduced a new evaluation methodology that has leveraged a dense set of correspondences generated from a multi-shot structured light system to map a thresholded camera pixel to the projector space. We have demonstrated that the proposed method maintains the error rate of the baseline state-of-the-art method, while achieves a close qualitative results without parameter fine tuning.

**Table 1:** Error rate, $\xi$, for the proposed method and the method in [8].

| Surface | Method | $\xi$ |
|---|---|---|
| *Wolf Head* | SP [8] | **0.01%** |
| | *Proposed* | 0.03% |
| *Zigzag* | SP [8] | **0.62%** |
| | *Proposed* | 0.64% |
| *Building Facade* | SP [8] | **0.58%** |
| | *Proposed* | 0.62% |

## References

[1] A. Gruen and E. P. Baltsavias, "Close-range photogrammetry meets machine vision," in *SPIE*, vol. 1395, 1990.

[2] J. Geng, "Structured-light 3d surface imaging: a tutorial," *Advances in Optics and Photonics*, vol. 3, no. 2, pp. 128–160, 2011.

[3] R. A. Morano, C. Ozturk, R. Conn, S. Dubin, S. Zietz, and J. Nissano, "Structured light using pseudorandom codes," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 20, no. 3, pp. 322–327, 1998.

[4] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE Trans. on Systems, Man, and Cybernetics*, vol. 9, no. 1, pp. 62–66, 1979.

[5] T. Pun, "A new method for grey-level picture thresholding using the entropy of the histogram," *Signal Processing*, vol. 2, no. 3, pp. 223–237, 1980.

[6] P. Lavoie, D. Ionescu, and E. M. Petriu, "3-d object model recovery from 2-d images using structured light," in *IEEE Instrumentation and Measurement Technology Conference and IMEKO Tec*, vol. 1, 1996, pp. 377–382 vol.1.

[7] W. Niblack, *An Introduction to Digital Image Processing*. DNK: Strandberg Publishing Company, 1985.

[8] J. Sauvola and M. Pietikäinen, "Adaptive document image binarization," *Pattern Recognition*, vol. 33, no. 2, pp. 225–236, 2000.

[9] D. Bradley and G. Roth, "Adaptive thresholding using the integral image," *J. Graphics Tools*, vol. 12, no. 2, pp. 13–21, 2007.

[10] S. Tang, X. Zhang, Z. Song, L. Song, and H. Zeng, "Robust pattern decoding in shape-coded structured light," *Optics and Lasers in Engineering*, vol. 96, pp. 50–62, 2017.

[11] K. Yang, Z. Ling, J. Li, X. Gao, L. Xie, and Z. Bai, "Color m-array shape reconstruction of using grid points and center points," in *Int. Conf. Inf. Optics and Photonics (CIOP)*, vol. 11209, 2019.

[12] G. Phonsa and K. Manu, "A survey: Image segmentation techniques," in *Harmony Search and Nature Inspired Optimization Algorithms*. Springer, 2019, pp. 1123–1140.

[13] M. Sezgin and B. Sankur, "Survey over image thresholding techniques and quantitative performance evaluation," *J. Electronic imaging*, vol. 13, no. 1, pp. 146–166, 2004.

[14] F. Li, H. Sekkati, J. Deglint, C. Scharfenberger, M. Lamm, D. Clausi, J. Zelek, and A. Wong, "Simultaneous projector-camera self-calibration for three-dimensional reconstruction and projection mapping," *IEEE Trans. on Computational Imaging*, vol. 3, no. 1, pp. 74–83, 2017.

[15] J. Salvi, J. Pagès, and J. Batlle, "Pattern codification strategies in structured light systems," *Pattern Recognition*, vol. 37, pp. 827–849, 2004.