Reconstructing Intensity Images from Binary Spatial Gradient Cameras

Suren Jayasuriya[1,2] Orazio Gallo[2] Jinwei Gu[2] Timo Aila[2] Jan Kautz[2]
[1]Carnegie Mellon University, [2]NVIDIA

Abstract

Binary gradient cameras extract edge and temporal information directly on the sensor, allowing for low-power, low-bandwidth, and high-dynamic-range capabilities—all critical factors for the deployment of embedded computer vision systems. However, these types of images require specialized computer vision algorithms and are not easy to interpret by a human observer. In this paper we propose to recover an intensity image from a single binary spatial gradient image with a deep autoencoder. Extensive experimental results on both simulated and real data show the effectiveness of the proposed approach.

1. Introduction

Gradient information, either temporal or spatial, has been widely used for a variety of computer vision algorithms from visual recognition, to feature detection, to optical flow and stereo reconstruction. Recently proposed computational cameras can calculate the image gradient directly on-chip, thus saving power and bandwidth as compared to regular CMOS image sensors. This is valuable for embedded vision applications, which have stringent power and bandwidth limitations for the image sensing stage. For instance, Google Glass operating a modern face recognition algorithm has a battery life of less than 40 minutes, with image sensing and computation each consuming roughly 50% of the power budget [19]; the cost of sending compressed images or video for off-line processing in the cloud is also several orders of magnitude higher than on-chip processing [22]. Gradient cameras represent a promising technology to overcome these limitations for embedded vision.

A popular type of binary, gradient camera is the dynamic vision sensor (DVS), which asynchronously outputs pixels recording a temporal change in intensity [18]. This camera has been successfully used for several traditional computer vision tasks.

For static scenes, however, a DVS camera does not capture any gradient information unless the camera moves. In this paper, we focus on binary *spatial* gradients, where only the pixels in high-contrast regions become active. The resulting images appear like binary edge images (Figure 1), and do not require any motion. These images are related to those produced by the DVS camera: the difference of two consecutive spatial gradient frames essentially produces a temporal gradient image.

Figure 1: Figure showing a captured binary spatial gradient video (left), our intensity reconstruction (middle), and the prototype camera we used for the capture.

Spatial binary gradients can be captured with specialized sensors that allow for a significant reduction of the power required to acquire, process, and transmit images [7]. However, the information they extract from the scene is limited. In this paper, we investigate whether the intensity information can be reconstructed from binary spatial gradient images in post-processing. This would be useful for tasks requiring a human in the loop, such as video surveillance on a limited power and bandwidth budget: a low-power system can run continuously, and when an event of interest is detected, a human observer inspects the intensity image. This data can be gathered by triggering a more power-hungry sensor [9], but it would be more efficient to extract it directly from the binary data itself. We argue that, in addition to reducing the bandwidth requirements, moving power consumption from sensing to post-processing scales better with Moore's law, as digital processing becomes cheaper and faster.

We show that intensity reconstruction from single-shot, spatial binary gradients, is indeed possible. An example of an image captured with a prototype camera, and the corresponding gray-scale image reconstructed with our approach

A version of the paper with animated figures can be found at the following link: *https://research.nvidia.com/publication/reconstructing-intensity-images-binary-spatial-gradient-cameras*.

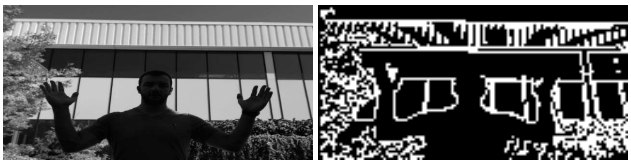


Figure 2: A traditional image (left) and an example of real spatial binary gradient data (right). Note that these pictures were taken with different cameras and lenses and, thus, do not exactly match.

are shown in Figure 1. To the best of our knowledge, we are the first to reconstruct intensity images from binary gradient data captured with this type of camera, in part because this is an ill-posed problem: both the direction and the sign of the gradient are lost (Section 3.1).

We utilize a deep autoencoder network to recover the missing intensity information. We perform our formal tests simulating the output of the sensor on existing datasets, but we also validate our findings by capturing real data with the prototype developed by Gottardi *et al*. [7], which implements this acquisition scheme. We believe that this paper presents a compelling reason for using binary spatial gradient cameras in certain computer vision tasks to reduce the power and bandwidth consumption for embedded systems.

2. Related Work

We describe the prior art in terms of the gradient cameras that have been proposed, and then in terms of computer vision algorithms developed for this type of data.

Gradient cameras can compute spatial gradients either in the optical domain [4, 31, 16], or on-board the image sensor, a technique known as focal plane processing [3, 17, 20, 10]. The gradients can be either calculated using adjacent pixels [7], or using current-mode image sensors [8]. Some cameras can also compute temporal gradient images, *i.e*. images where the active pixels indicate a temporal change in local contrast [7, 18]. Most of these gradient cameras have side benefits of fast frame rates and reduced data bandwidth/power due to the sparseness of gradients in a scene. In fact, the camera by Lichtsteiner *et al*. can read individual pixels when they become active [18]. Moreover, the fact that gradient cameras output a function of the difference of two or more pixels, rather than the pixel values themselves, allows them to deal with high-dynamic-range scenes.

Applications of gradient cameras were first exposited in the work by Tumblin *et al*., who described the advantages of reading pixel differences rather than absolute values [25]. The appealing benefits of gradient cameras spurred the interest of the computer vision community, which adapted a number of traditional techniques to this new type of data. For instance Weikersdorfer *et al*. proposed to use SLAM with DVS cameras [27], and O'Connor *et al*. coupled them with spiking networks for real-time classification [21]. Another area that recently received a great interest is that of intensity reconstruction from sparse gradient data. This is often coupled with a vision task: Kim *et al*. proposed a method to perform simultaneous intensity reconstruction and object tracking [13], Bardow *et al*. combined optical flow and intensity reconstruction [1], Barua *et al*. did face detection and intensity reconstruction [2], and Kim *et al*. performed simultaneous depth, localization, and intensity reconstruction [14].

The intensity reconstruction offered by these methods is impressive, but requires two assumptions. First, the camera, the scene, or both must be dynamic: the sensor does not output any information otherwise. Second, several consecutive frames must be available to perform the reconstruction: we are not aware of any method that can perform intensity reconstruction from a single frame.

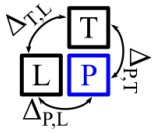
In contrast, we focus on spatial binary gradients, which work for static scenes as well as dynamic ones. Our method can reconstruct intensity images from a single binary gradient frame.

3. Method

In this section, we first outline the binary spatial gradient camera's operation based on the sensor by Gottardi *et al*. [7], which we use as a reference implementation for a system that captures this type of data. We then describe our reconstruction approach, and show results on data we generated with our simulator. Finally we verify our findings on data captured with a real prototype.

3.1. Background: Operation and Power Estimate

With spatial binary gradients, we refer to cameras for which a pixel becomes active when a local measure of contrast is above threshold. Specifically, for two pixels i and j, we define the difference $\Delta_{i,j} = |I_i - I_j|$, where I is the measured pixel's brightness. We also define a neighborhood ν consisting of pixel P and the pixels to its left, L, and top, T (see inset). The output at pixel P will then be:



$$G_S(\mathrm{P}) = \begin{cases} 1 & \text{if } \max\limits_{i,j \in \nu} \Delta_{i,j} > T \\ 0 & \text{otherwise} \end{cases}, \qquad (1)$$

where T is a threshold set at capture time. The output of this operation is a binary image where changes in local spatial contrast above threshold yield a 1, else a 0, see Figure 2.

Note that Equation 1 is an approximation of a binary local derivative: $\Delta_{T,L}$ alone can trigger an activation for P, even though the intensity at P is not significantly different

from either of the neighbors'. It can be shown that the consequence of this approximation is a "fattening" of the image edges by a factor of roughly $\sqrt{2}$ when compared to the magnitude of the a gradient computed with regular finite differences.

Also, because the sign of the derivative is lost, a dark object against a bright background would yield the same exact binary spatial gradient as a bright object on a dark background. In the context of reconstructing the intensity image, this ambiguity prevents the methods of surface integration from working, even with known boundary conditions.

The advantage of this formulation is that it can be implemented efficiently in hardware, leading to significant power savings. The power consumption for the sensor by Gottardi *et al*. [7] can be approximated by the sum of two components. The first, independent of the actual number of active pixels, is the power required to scan the sensor and amounts to 0.0024μW/pixel. The second is the power required to deliver the addresses of the active pixels, and is 0.0195μW/pixel [6]. While the number of active pixels is a function of the scene, Gottardi *et al*. [7] report that for typical scenes this number is usually below 25% (in the data we captured, we actually measured that slightly less than 10% of the pixels were active on average). At 30fps, this power corresponds to 7.3pJ/pixel. A modern image sensor, for comparison, is over 300pJ/pixel [24]. While these numbers are to be taken as rough estimates, they do offer an insight on the power savings that one can reasonably expect.

3.2. Recovering Intensity Information from Binary Spatial Gradients

We take a deep learning approach to intensity reconstruction and, specifically, we use an autoencoder (AE) architecture, see Figure 3. Autoencoders learn a lower-dimensional representation of a signal, and thus are particularly well-suited to learn priors on the distribution of the data and the noise [26]. This is a very attractive feature that helps with our problem being intrinsically ill-posed (Section 3.1).

In our experiments, we sought to find a compromise between the AE's depth and the accuracy of the results. The resulting architecture comprises 5 subsampling units for the encoding stage, each followed by a max-pooling layer. The decoding stage is symmetric with 5 units, each followed by upsampling instead of max-pooling. For both the encoding and the decoding stages, these units consists of 2 convolutional layers and leaky ReLUs. Finally every convolutional layer consists of 100 filters of kernel size 3×3.

Because AEs significantly downsample the data, they sometimes produce blurry results. We addressed this problem by using skip-connections [11], which propagate high-frequency information directly to the decoding stage from the appropriate encoding unit.

The loss function has a strong impact on the quality of the results [30]. Therefore, we tested several loss functions including ℓ_2, ℓ_1 and total variation regularization. For the purpose of our reconstruction, these alternative losses did not produce better results than those produced by ℓ_1, which is why we chose it as our loss function.

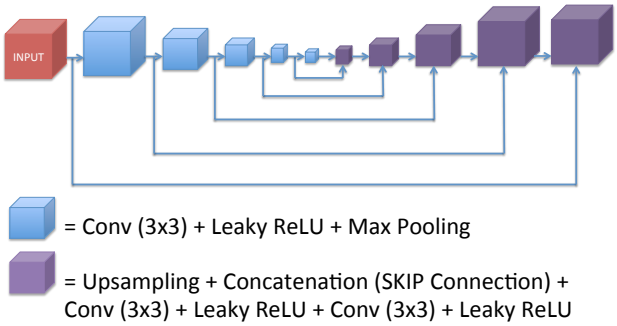


Figure 3: The architecture of the autoencoder used to reconstruct intensity information from spatial binary gradient images.

Finally, we used ADAM [15] for optimization with learning rate $\lambda = 0.00001$, and found that the learning schedule did not have a significant impact on convergence.

3.3. Training Data

To generate a sufficiently large amount of data to train our AE, we wrote a simulator of the camera described in Section 3.1. We empirically tuned the threshold T to match the appearance of the simulated and real data, which we captured with a prototype camera. T typically varied from 0.05 to 0.1 for pixel intensities normalized to one. We were then able to leverage existing datasets to create pairs of gradient/ground truth images.

We used TensorFlow and Keras to construct our networks. All experiments were performed on a cluster of GPUs with NVIDIA Titan X's or K80s.

We trained our AE on two datasets for faces, BIWI [5] and WIDER [28], and also the LSUN [29] dataset for indoor and outdoor scenes.

BIWI: The BIWI face dataset, contains 15,000 images of 20 subjects, each accompanied by a depth image, as well as the head 3D location and orientation [5]. We removed two subjects completely to be used for testing.

WIDER: We also trained the network on the WIDER face dataset, a collection of 30,000+ images with 390,000+ faces [28]. The WIDER dataset, does not contain repeated images of any one person except for a few celebrities which we remove from our testing set, guaranteeing that no test face is seen by the network during training. We extracted face crops by running a face detection algorithm [23], and resized them to 96x96, by either downsampling or upsampling, unless the original size was too small.

LSUN: The LSUN dataset is an extremely large collection of images divided in several categories [29]. We used it to verify the ability of our network architecture to learn data from a more diverse distribution. We focused on one indoor ('bedroom') and one outdoor ('church') category. We use roughly 100k images for each dataset. Note that the size of the dataset comes at a cost: there are several outlier images, and many of the other images have artifacts, such as watermarks or overlaid text.

4. Experimental Results

We reconstruct intensity images from both simulated data, and from the *actual* output of a prototype camera that implements the algorithm described in Section 3.1.

Regarding the simulated data, a note on the threshold T used in Equation 1 is in order. Similarly to exposure time for traditional cameras, T should be adapted to the content of the image so that the binary gradients are not too sparse, Figure 6(j), nor too crowded, Figure 7(b). However, instead of defining an arbitrary algorithm to select T for each image independently, we opted to empirically set a unique threshold for each one of the datasets. The quality of the reconstruction degrades when T significantly deviates from its optimal value, leading us to believe that a per-image threshold would only improve the results. Below we report the threshold we used for each dataset.

4.1. Simulated Data

Figure 4 shows the reconstruction on one of the two testing subjects from the BIWI dataset. The threshold T used was 0.05 for this dataset. As mentioned above, the solution is not unique given the binarized nature of the gradient image, and indeed the network fails to estimate the shade of the first subject's sweater. Nevertheless, the quality is sufficient to identify the person in the picture, which is surprising, given the sparseness of the input data.

Figure 5 shows results of the reconstruction for the WIDER dataset. The threshold T used was 0.09. Note that the failure cases are those where the quality of the gradients is not sufficient, Figure 5(i), or the face is occluded, Figure 5(j). The rest of the faces are reconstructed well, once again, allowing to identify the person.

Figure 6 shows some reconstructed images for the 'bedroom' category of the LSUN dataset. The threshold $T = 0.07$ for this dataset. This dataset presents a more significant variability in terms of the actual image content, see for instance Figure 6(j). This weakens the prior on the expected image content. Nevertheless, our network produces reasonable reconstructions whether the input portraits a relatively standard bedroom setup, or when it contains less common subjects, such as kids (i) or even a cat (j).

Figure 7 shows some reconstructed images for the 'church' category of the LSUN dataset. The threshold $T = 0.07$ for this dataset. This is probably the most difficult dataset for the network due to the variability of the data. Also the assumption that a single threshold can be used for the whole dataset works more poorly due to the varying dynamic range of different images, causing several binary gradient images to be overly-active. This reflects in a poorer quality of the reconstruction.

Figure 4: Figure of the intensity reconstruction (middle pane) on the binary data (left pane) simulated from the BIWI dataset [5]. The ground truth is on the right.

4.2. Real Data

We validate our findings by running experiments on binary gradient images captured with the actual prototype camera described by Gottardi *et al*. [7]. The spatial resolution of this camera is 128x64 pixels, which limits the quality of the spatial gradients. We use the widest aperture setting allowed by the lens to gain the most light, though at the cost of a shallower depth of field, which we did not find to affect the quality of the gradient image. To validate our simulator, we also captured a few grayscale images of the same scene with a second camera set up to roughly match the field of views of the two. Figure 2, shows a comparison between a grayscale image and the (roughly) corresponding frame from the prototype camera. Barring resolution issues, we believe our simulations match the real data (compare for instance the real data in Figure 5 and the simulated data in the second row of Figure 8).

We trained the network on synthetic data generated from the WIDER dataset at the resolution of 64x64. We then performed forward inference on the real data. We did not perform fine-tuning due to the lack of ground truth data—the data from an intensity camera captured from a slightly different position, and with different lenses, did not generalize well. While the quality of the reconstruction is slightly degraded with respect to that of the synthetic data, the faces are reconstructed well. Figure 1 shows an animation on a captured binary gradient video. We are reconstructing intensity information from a single frame: we are not enforcing temporal consistency, nor we use information from multiple frames to better infer intensity. Figure 8 shows a few static frames from different subjects. Note that despite the low resolution (these crops are 1.5 times smaller than those in Figure 5), the face features are still recognizable.

Figure 5: Intensity reconstruction (bottom row) on the binary data (middle row) simulated from the WIDER dataset [28]. The ground truth is in the top row. Note that our neural network is able to recover the fine details needed to identify the subjects. We observed that failure cases happen when the gradients are simply too poor (i) or the face is occluded (j).

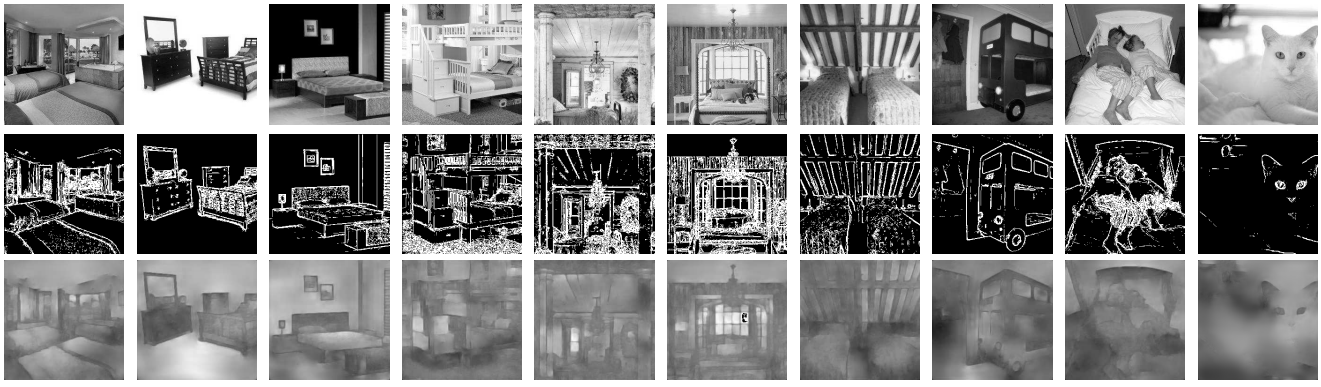


Figure 6: Intensity reconstruction (bottom row) on the binary data (middle row) simulated from the 'Bedroom' category from the LSUN dataset [29]. The ground truth is in the top row. The dataset is very large, but the images it contains are not very carefully chosen, thus allowing for spurious subjects, such as the cat in (j). However, even in such cases, our network produces a reasonable reconstruction so long as the gradients are correctly captured, such as the kids in (i) or cat in (j).

We find that the quality of the reconstruction of any single frame varies: some reconstructions from real data allow the viewer to determine the identity of the subject, others are more similar to average faces.

5. Discussion and limitations

We believe that our intensity reconstruction results are good, but they can still be improved. While the ability to reconstruct intensity from a single image is important, incorporating temporal information may be beneficial. This could be achieved by using a recurrent network that works on sequences of frames. Note that this is different from using temporal gradients, though the network can learn to generate them, because of the issues with static and dynamic content. Our reconstruction leverages the fact that we can learn a prior about the data. This could lead to failure when the scene is significantly different from what the network was trained on, although we have observed that the network still produces reasonable results even in those circumstances (as seen in Figure 6(i)). Finally, aside from trying to learn the more likely color from the training data, the network cannot disambiguate the intrinsic ill-posedness of the binary gradient data.

6. Conclusion

We have proposed to use an autoencoder network to learn the prior distribution of a specific class of images to solve the under-constrained problem of recovering intensity information from binary spatial edges. We are able to achieve visually plausible reconstructions for several classes of images simulating the binary gradient data on existing image datasets. We also validate our method on

Figure 7: Intensity reconstruction (bottom row) on the binary data (middle row) simulated from the 'Church' category from the LSUN dataset [29]. The ground truth is in the top row.

Figure 8: Intensity reconstruction result inferred by the network described in Section 3.2 and trained on the WIDER simulated data. The top row shows 64x64 face crops captured with the prototype camera, the bottom the corresponding reconstructed images. While the quality is not quite on par with the intensity reconstructions, it has to be noted that the resolution of the crops in Figure 5, is 96x96, *i.e*. 1.5x larger.

real images taken from a prototype camera.

There are several avenues for future research in gradient cameras. Recent work in GANs has shown success in converting sketches or edge maps to full colored images [12] which may yield better intensity reconstructions for our problem. In addition, collecting a large dataset of registered RGB and gradient images would enable better machine learning and allow us to investigate the use of gradient cameras for classification and other visual recognition tasks.

Acknowledgements

We would like to thank Massimo Gottardi from the Fondazione Bruno Kessler for loaning to us the prototype camera used in this work. We would also like to acknowledge Pavlo Molchanov and Alejandro Troccoli who lent us their faces for the reconstruction experiments.

References

[1] P. Bardow, A. J. Davison, and S. Leutenegger. Simultaneous optical flow and intensity estimation from an event camera. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[2] S. Barua, Y. Miyatani, and A. Veeraraghavan. Direct face detection and video reconstruction from event cameras. In *Applications of Computer Vision (WACV), 2016 IEEE Winter Conference on*, pages 1–9. IEEE, 2016.

[3] S. M. Chai, A. Gentile, W. E. Lugo-Beauchamp, J. Fonseca, J. L. Cruz-Rivera, and D. S. Wills. Focal-plane processing architectures for real-time hyperspectral image processing. *Applied Optics*, 39(5):835–849, 2000.

[4] H. G. Chen, S. Jayasuriya, J. Yang, J. Stephen, S. Sivaramakrishnan, A. Veeraraghavan, and A. Molnar. ASP vision: Optically computing the first layer of convolutional neural networks using angle sensitive pixels. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[5] G. Fanelli, M. Dantone, J. Gall, A. Fossati, and L. Van Gool. Random forests for real time 3d face analysis. *Int. J. Comput. Vision*, 101(3):437–458, February 2013.

[6] O. Gallo, I. Frosio, L. Gasparini, K. Pulli, and M. Gottardi. Retrieving gray-level information from a binary sensor and its application to gesture detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 21–26, 2015.

[7] M. Gottardi, N. Massari, and S. Jawed. A 100 μW 128$\times$64 pixels contrast-based asynchronous binary vision sensor for sensor networks applications. *IEEE Journal of Solid-State Circuits*, 44(5):1582–1592, 2009.

[8] V. Gruev, R. Etienne-Cummings, and T. Horiuchi. Linear current mode imager with low fix pattern noise. In *Circuits and Systems, 2004. ISCAS'04. Proceedings of the 2004 In-*

ternational Symposium on, volume 4, pages IV–860. IEEE, 2004.

[9] S. Han, R. Nandakumar, M. Philipose, A. Krishnamurthy, and D. Wetherall. Glimpsedata: Towards continuous vision-based personal analytics. In *Proceedings of the 2014 workshop on physical analytics*, pages 31–36. ACM, 2014.

[10] P. Hasler. Low-power programmable signal processing. In *Fifth International Workshop on System-on-Chip for Real-Time Applications (IWSOC'05)*, pages 413–418. IEEE, 2005.

[11] E. Ilg, N. Mayer, T. Saikia, M. Keuper, A. Dosovitskiy, and T. Brox. Flownet 2.0: Evolution of optical flow estimation with deep networks. *arXiv preprint arXiv:1612.01925*, 2016.

[12] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros. Image-to-image translation with conditional adversarial networks. *arXiv preprint arXiv:1611.07004*, 2016.

[13] H. Kim, A. Handa, R. Benosman, S.-H. Ieng, and A. J. Davison. Simultaneous mosaicing and tracking with an event camera. *J. Solid State Circ*, 43:566–576, 2008.

[14] H. Kim, S. Leutenegger, and A. J. Davison. Real-time 3d reconstruction and 6-dof tracking with an event camera. In *European Conference on Computer Vision*, pages 349–364. Springer, 2016.

[15] D. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[16] S. J. Koppal, I. Gkioulekas, T. Young, H. Park, K. B. Crozier, G. L. Barrows, and T. Zickler. Toward wide-angle microvision sensors. *IEEE transactions on pattern analysis and machine intelligence*, 35(12):2982–2996, 2013.

[17] W. D. Leon-Salas, S. Balkir, K. Sayood, N. Schemm, and M. W. Hoffman. A cmos imager with focal plane compression using predictive coding. *IEEE Journal of Solid-State Circuits*, 42(11):2555–2572, 2007.

[18] P. Lichtsteiner, C. Posch, and T. Delbruck. A 128× 128 120 db 15 μs latency asynchronous temporal contrast vision sensor. *IEEE journal of solid-state circuits*, 43(2):566–576, 2008.

[19] R. LiKamWa, Z. Wang, A. Carroll, F. X. Lin, and L. Zhong. Draining our glass: An energy and heat characterization of google glass. In *Proceedings of 5th Asia-Pacific Workshop on Systems*. ACM, 2014.

[20] A. Nilchi, J. Aziz, and R. Genov. Focal-plane algorithmically-multiplying cmos computational image sensor. *IEEE Journal of Solid-State Circuits*, 44(6):1829–1839, 2009.

[21] P. O'Connor, D. Neil, S.-C. Liu, T. Delbruck, and M. Pfeiffer. Real-time classification and sensor fusion with a spiking deep belief network. *Neuromorphic Engineering Systems and Applications*, page 61, 2015.

[22] J. M. Ragan-Kelley. *Decoupling algorithms from the organization of computation for high performance image processing*. PhD thesis, Ch. 2, pages 19–24, Massachusetts Institute of Technology, 2014.

[23] S. Ren, K. He, R. Girshick, and J. Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems (NIPS)*, 2015.

[24] A. Suzuki, N. Shimamura, T. Kainuma, N. Kawazu, C. Okada, T. Oka, K. Koiso, A. Masagaki, Y. Yagasaki, S. Gonoi, et al. A 1/1.7-inch 20mpixel back-illuminated stacked CMOS image sensor for new imaging applications. In *2015 IEEE International Solid-State Circuits Conference-(ISSCC) Digest of Technical Papers*, pages 1–3. IEEE, 2015.

[25] J. Tumblin, A. Agrawal, and R. Raskar. Why i want a gradient camera. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 1, pages 103–110. IEEE, 2005.

[26] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, and P.-A. Manzagol. Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *Journal of Machine Learning Research*, 11(Dec):3371–3408, 2010.

[27] D. Weikersdorfer, R. Hoffmann, and J. Conradt. Simultaneous localization and mapping for event-based vision systems. In *International Conference on Computer Vision Systems*, pages 133–142. Springer, 2013.

[28] S. Yang, P. Luo, C. C. Loy, and X. Tang. Wider face: A face detection benchmark. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[29] F. Yu, Y. Zhang, S. Song, A. Seff, and J. Xiao. Lsun: Construction of a large-scale image dataset using deep learning with humans in the loop. *arXiv preprint arXiv:1506.03365*, 2015.

[30] H. Zhao, O. Gallo, I. Frosio, and J. Kautz. Loss functions for image restoration with neural networks. *IEEE Transactions on Computational Imaging*, 2016.

[31] A. Zomet and S. K. Nayar. Lensless imaging with a controllable aperture. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, volume 1, pages 339–346. IEEE, 2006.