

Detection of laser-induced optical defects based on image segmentation

Xinkun Chu¹, Hao Zhang¹, Zhiyu Tian¹, Qing Zhang¹, Fang Wang², Jing Chen², and Yuanchao Geng²

¹Institute of Computer Application, China Academy of Engineering Physics, Mianyang 621900, China

²Research Center of Laser Fusion, China Academy of Engineering Physics, Mianyang 621900, China

(Received 25 June 2019; revised 20 October 2019; accepted 9 November 2019)

Abstract

A number of vision-based methods for detecting laser-induced defects on optical components have been implemented to replace the time-consuming manual inspection. While deep-learning-based methods have achieved state-of-the-art performances in many visual recognition tasks, their success often hinges on the availability of a large number of labeled training sets. In this paper, we propose a surface defect detection method based on image segmentation with a U-shaped convolutional network (U-Net). The designed network was trained on paired sets of online and offline images of optics from a large laser facility. We show in our experimental evaluation that our approach can accurately locate laser-induced defects on the optics in real time. The main advantage of the proposed method is that the network can be trained end to end on small samples, without the requirement for manual labeling or manual feature extraction. The approach can be applied to the daily inspection and maintenance of optical components in large laser facilities.

Keywords: deep learning; defect detection; laser-induced defects

1. Introduction

Defects on the surface of optics are among the earliest indications of degradation which are critical for the maintenance of optical systems. Early detection of the defects allows preventive measures to be taken to prevent the defects from growing to an unrepairable size. Large laser facilities, such as the National Ignition Facility (NIF)^[1] and the Laser Megajoule (LMJ)^[2], routinely operate at high ultraviolet fluences above the damage threshold of optical components. The laser-induced defects on optics, once initiated, will grow rapidly in subsequent exposure to high fluence, until to the point at which the entire optical component needs to be replaced. Therefore, it is critical for sustainable operation to detect and monitor defects in the early stage.

Various image processing techniques, such as the threshold method, Otsu's method and Fourier transform^[3–5], have been implemented for defect detection to replace the time-consuming and error-prone manual inspection. Scientists at the Lawrence Livermore National Laboratory (LLNL) have conducted a lot of valuable researches in the field of damage online inspection. Using linescan phase-differential imaging, LLNL developed a process for rapid detection of phase defects in the bulk or surface of large-aperture

optics^[6]. A threshold is set on the brightest pixel value to select candidates for further assessment of their fratricidal threat. LLNL also designed the local area signal-to-noise ratio (LASNR) algorithm^[7] for accurate and rapid inspection of the optics from the NIF. The algorithm estimates the strength of signal within an object versus the noise in its local neighborhood. However, the accuracy and robustness of these image processing techniques are largely affected by varying situations like illumination conditions, shading and noises.

Machine-learning-based models outperform the image processing techniques in accuracy and robustness, and have been successfully applied in computer vision tasks such as object detection and classification. LLNL extracted various features from each damage site and employed ensemble of decision trees to identify false damage sites from hardware reflections^[8]. Harbin Institute of Technology (HIT) developed the final optics damage inspection (FODI) system for the laser facility at the China Academy of Engineering Physics (CAEP)^[9, 10]. HIT manually extracted features associated with each damage site, and then used extreme learning machine to distinguish true and false damage sites and predict the damage size. The success of the machine learning models above relies on the manually custom-built features based on the experience of domain experts. Nathan *et al.*^[11] built convolutional neural network

Correspondence to: Y. Geng, No. 64 Mianshan Road, Mianyang 621900, China. Email: gengyuanchao@caep.cn

(CNN) with features auto-generated from data, and found that it outperforms the method using custom-built features in finding the subtle, rare, incomplete repairs of damage. However, typical CNNs rely heavily on the availability of a large labeled training sample. The label for each potential damage site is identified by a microscope. It is difficult and time-consuming to obtain a large labeled sample in practice for training the models.

In recent years, fully convolutional networks with U-shaped architecture (U-Net) have been lauded for precise segmentation and efficient use of available samples. Models based on U-Net are commonly used in image analysis tasks in medical diagnosis, biological science and cosmology^[12–15]. Dong *et al.*^[16] used features sampled from the earlier U-Net layers to identify defects in radiographs of aerospace welds. Inspired by these works, we present our approach to detection of optical defects that leverages the power of U-Net. We trained the network end to end on paired sets of online and offline images of optics from a large laser facility. The main advantage of the proposed method is that the network can be trained from very few images without manual labeling or manual feature extraction.

The paper is outlined as follows. First, we introduce the structure of the detection model based on U-Net. Then, we explain in detail about the methodology used in building the model, including the overall architecture, the preparation of the training set, the specifics of its implementation and training procedure. Finally, we show the robustness and adaptability of the model for online detection on the laser facility using novel optical images never seen by the network.

2. U-Net for defect detection

Convolutional neural networks serve as the network backbones for image segmentation due to the high representation power and filter sharing properties. The U-Net architecture is built upon the fully convolutional networks. It consists of a contracting path to capture context and a symmetric expanding path that enables precise localization. The schematic representation of our model used for defect detection is shown in Figure 1, which follows the original architecture proposed by Ronneberger *et al.* in 2015^[12]. The dimensions, padding method and other parameters of the model are adjusted to make the model better suited to our task.

The network consists of four major operations, convolution, up-convolution, max pooling and feature forwarding, as shown by the arrows in Figure 1. The operation of convolution processes an image using a receptive field that detects specific attributes like lines and edges. More complex attributes are detected as the layers go deeper. An up-convolution is the transpose matrix operation of the normal convolution, which remaps the detected attributes into the image space. Pooling layers are inserted between the successive convolutional layers to reduce the amount

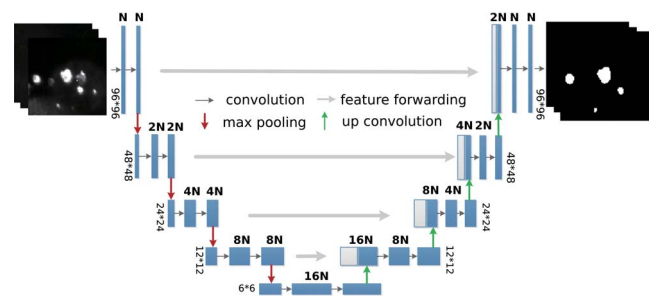


Figure 1. Schematic representation of the proposed U-Net model for defect detection. The boxes correspond to multi-channel feature maps, connected by different operations (denoted by arrows). The length and height of each box represent the number of filters ($N = 32$) and the x - y size, respectively.

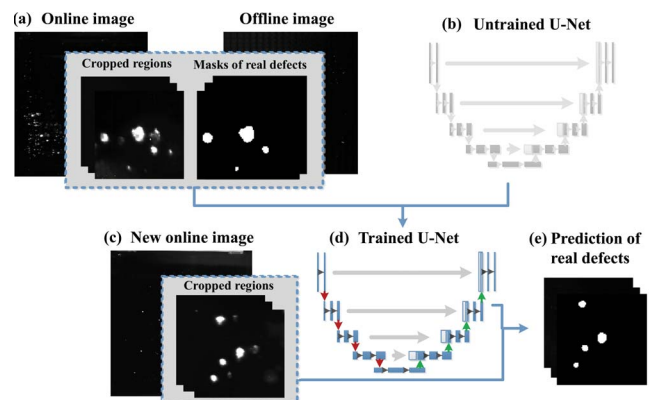


Figure 2. The overall architecture to train the model for detection of optical defects in real time.

of parameters and computational cost. The high-resolution features from the contracting path on the left are forwarded to the expanding path by combining with the up-sampled output. In addition, dropout layers are added at the end of the contracting path to prevent the model from over-fitting. The final convolutional layer's output has the same dimension as the input. A sigmoid function is applied to convert the output to the probability of real laser-induced defects.

3. Methodology

3.1. Overall architecture

Figure 2 shows the overall architecture of our method to train the model for online detection of optical defects. Raw images of optics were taken from the monitoring system of a large laser facility, as described in detail in Ref. [9]. The training set consists of pairs of online regions and corresponding masks of real defects (Figure 2(a)). The masks were created by the images of optics taken offline. Figure 2(b) is the unfitted U-Net model with untrained parameters. The structure of the model has been shown in Section 2. We trained the model using the training set

to learn the mapping from the online optics to the masks of real defects. To test whether the system could make accurate predictions for novel online images, the trained U-Net (Figure 2(d)) was used to locate defects for images from different beamlines. The input (Figure 2(c)) of the trained U-Net is thus the online image of the optic. The output (Figure 2(e)) is a mask whose value is the network’s certainty that an input element is a part of real defects.

3.2. Training set preparation

To train the U-Net to detect defects from online images, we first prepared a dataset of training samples, consisting of pairs of regions from online images and corresponding masks created by the offline images. The online images of the optics were acquired using the camera system placed at the center of the target chamber between the laser shots. Defects on the optics scatter light into the CCD yielding bright signals against dark backgrounds. However, a potential defect site on the online image can fall into one of several categories, such as real laser-induced defect, hardware reflection, light spot, reflection from the exit surface or damaged CCD pixels. Figure 3 shows examples of some categories to illustrate the difficulty in distinguishing between them.

Most of the optics on site are under daily maintenance. It will take long to accumulate sufficient number of defects for training the network. We selected two badly damaged optics after high exposure to the laser, which contained several hundreds of real laser-induced defects. After the online images were taken, the optics were disassembled from the frame and passed through a cleaning system. Then the offline images were collected by scanning the cleaned optic in a non-disturbing light environment. The offline images only contained laser-induced defects, without reflections, light spots and other on-site noises; hence, it can be used as the mask of real laser-induced defects for the online image. Figure 4 is the schematic of the experimental setup to illustrate the methodology in obtaining the online and offline images.

To determine the mapping between the online and offline images, a frame of reference was established by applying fiducials making up groups of small dots at the four corners of each optic. The circle Hough transform (CHT)^[17], as a feature-extraction technique for detecting circles, was further applied to add additional obvious matching points between the online and offline images of the inspected optic. A 3×3 matrix, called homography^[18], was then calculated based on these matching points that warped the offline image to the online image.

The resolution of the online and transformed offline images was around 3000×3000 , which was too large for the input of the neural network. The online images were cropped into small images of 96×96 pixel resolutions with

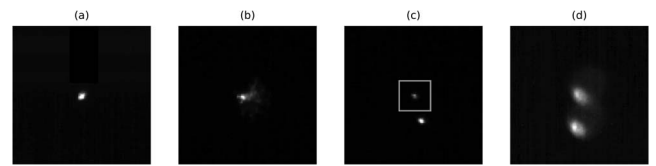


Figure 3. Examples of a potential damage site classified as: (a) real defect; (b) hardware reflection; (c) reflection from the exit surface (marked in the box); (d) light spot.

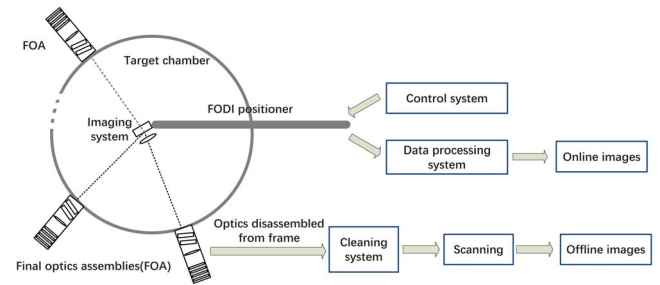


Figure 4. Schematic diagram of the methodology in obtaining the online and offline images of the final optics.

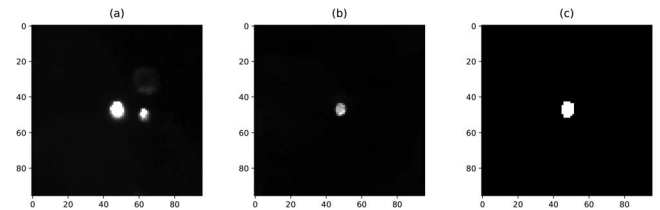


Figure 5. An example of the prepared training dataset: (a) the cropped region from the online image; (b) the matched region of (a) in the offline image; (c) the 0–1 mask created by (b), with 1 for real defect and 0 for background.

a stride of 48. Considering the pixel-level distortion and shift between the online and offline images, we obtained the matched offline region by searching the corresponding neighborhood for each cropped region using the template matching algorithm^[19]. We compared the online regions (I) against the offline regions (T) of 144×144 by sliding and computed a metric for each location^[20]. The location with the highest match value and the corresponding region of T were selected to get the matched offline region.

The LASNR algorithm was applied to mark the position of defects on the offline images and find the full extent of each defect. All the marked sites on the offline image could be considered as real defects; hence, 0–1 mask was given for each pixel, with 1 for real defect and 0 for background. Figure 5 shows an example of the prepared dataset for illustration. The inputs to the network were the cropped online images with 96×96 pixel resolutions, and the target outputs were the corresponding masks of the same dimension.

3.3. Implementation and training

Our implementation was realized in Python 3.6, using the Keras^[21] framework with TensorFlow backend. TensorFlow^[22] is a popular open-source library for deep learning, and Keras is a high-level neural networks library with a focus on enabling fast experimentation. All the training and experiments were run on a standard workstation with the Nvidia Tesla P100 GPU.

The total number of paired images for training the network was 550 (with 96×96 pixel resolutions for both the online samples and masks), after removal of images with all dark backgrounds. The samples were split randomly with a ratio of 4:1, yielding 440 training samples and 110 validation samples. The validation samples were used for early stopping to prevent the network from over-fitting when the network started to model the noise in the training samples.

For our task, the available training sample was quite small. Data augmentation was essential to teach the network the desired invariance. Morphology transformations like rotation, shift in width and height, horizontal and vertical flips, and variation in gray values were applied to images and masks at the same time. The data samples and parameters for augmentation were wrapped in a data generator, which generated batches of tensor image data for each training epoch.

The intensity distribution of the images was highly imbalanced, as shown in Figure 6. The dark backgrounds converged near 0 while the potential defects spread from 0 to 1.0 with a fraction of less than 3%. The bin near 1.0 was caused by over-exposure of some of the potential defects. The learning process was prone to get trapped in local minima of the loss function with predictions strongly biased toward 0. To solve the problem, we implemented a dice loss function in Keras following Milletari *et al.* (2016)^[23]. The dice coefficient (DCE) between two regions is defined as

$$\text{DCE} = \frac{2\vec{p} \cdot \vec{g}}{\vec{p} + \vec{g}}, \quad (1)$$

where \vec{p} and \vec{g} are vectors of the predicted results and the ground truth, respectively. We trained the network by minimizing the dice loss, which is defined as $1 - \text{DCE}$. Alternatively, we trained the network using the binary cross-entropy loss and assigned more weights to the foreground. It was found that training with the dice loss outperforms the re-weighting method and the training process could quickly pass the local minimum.

We trained the network using the Adam optimizer. The Adam optimization is an extension to stochastic gradient descent that can be used to update the network weights. The initial learning rate was 10^{-3} , which allowed the network to converge fast. The network was trained iteratively with a mini-batch of size 50 until the loss of the validation samples

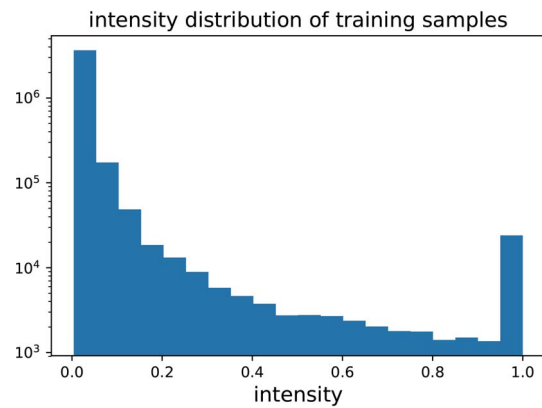


Figure 6. Intensity distribution of the training samples (in log scale).

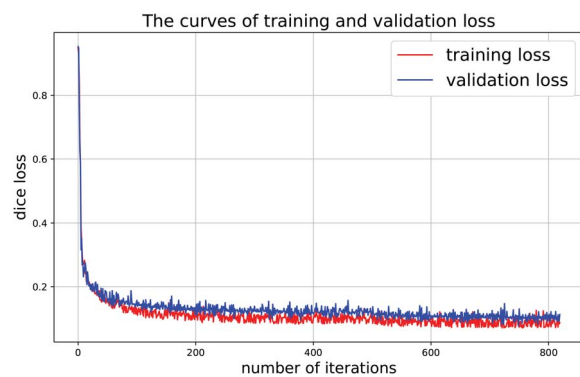


Figure 7. The curves of training and validation loss with respect to the number of iterations. We used a learning rate of 10^{-3} for the first 500 iterations and changed the learning rate to 10^{-4} for later iterations.

did not decrease. For each iteration, we fed randomly augmented samples as input to the network. Then, we changed the learning rate to 10^{-4} and iterated the above process for fine-tuning of the model. Figure 7 shows the curves of training and validation loss with respect to the number of iterations.

4. Results

To test the robustness and adaptability of the model for online detection, we took images of optics from different beamlines and prepared the testing set following the same method used to produce the training set. The online images were cropped into small regions of 96×96 pixel resolutions and fed as input to the trained U-Net model. Then the predictions by the model were used to locate the real laser-induced defects on the online image and get the predicted mask. The total number of testing samples was 636 (with 96×96 pixel resolutions). The sample contained 415 real laser-induced defects. The trained U-Net model achieved a DCE of 86.2% on the testing set. Figure 8 shows the predictions of defects by the trained model (Figure 8(c)),

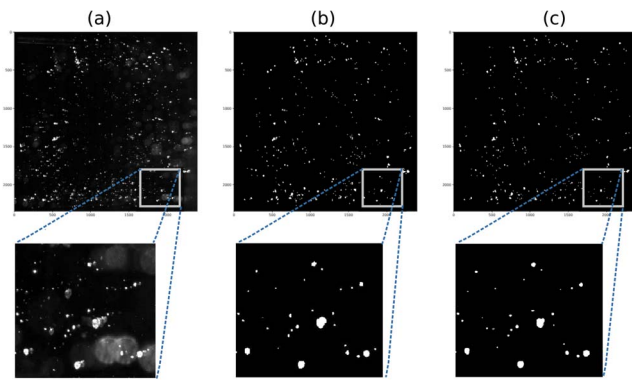


Figure 8. Predictions of real defects by the trained model on the test images. (a) The online image of an inspected optic. (b) 0–1 mask created by the offline images of the same inspected optic. (c) Predicted mask by the trained U-Net model. Bottom panels show a zoom-in on a highly contaminated region.

compared with the mask of real defects created by the offline image (Figure 8(b)), on one of the inspected optics (Figure 8(a)) as an example. As shown by the bottom panels of the figure, the trained model could separate real defects from backgrounds such as hardware reflections and light spots.

To further characterize the performance of the trained U-Net model, we calculated the precision (P) and recall (R) in object level, as given in terms of the number of true positives (TP), false positives (FP) and false negatives (FN):

$$P = TP / (TP + FP), \quad (2)$$

$$R = TP / (TP + FN), \quad (3)$$

where TP in our case corresponds to real defects correctly marked by the prediction of the model, FP/FN corresponds to false/real defects incorrectly marked as real/false defects by the model. The recall was 95.7% and the precision was 92.5% for the trained U-Net on the testing set. The FP/FN mainly came from some tiny defects of several pixels contaminated with backgrounds. And the F_1 score, defined as

$$F_1 = 2PR / (P + R), \quad (4)$$

was 0.94. We implemented the algorithm on a platform with user interface. It took less than 30 s to process an entire online image of 3000×3000 pixels on the platform, which met the system’s requirement for online detection between the laser shots on the laser facility.

5. Conclusion

In this paper, a vision-based approach for detecting optical defects has been proposed based on image segmentation. The proposed deep learning system can accurately locate

laser-induced defects on the optics in real time. Unlike typical classification models where the output to an image is a single label, the U-Net model is able to assign a class label to each pixel. Moreover, the detection model can be trained end to end on small samples without the requirement for manual labeling or manual feature extraction. The proposed method is especially strong at detecting defects when each sample may contain multiple adjacent objects. In our case, the model removes the fake defects from reflections by learning the relative spatial and intensity information, where we had limited success with typical classification models in previous studies. The proposed approach may have wide applications in the online detection and maintenance of large laser facilities where a large number of labeled samples are not available.

Nevertheless, we encountered some limitations of the current method. First, it is assumed that the object inspected does not have complicated structures. Hence, the offline images can be used as the mask of real defects. Second, the network’s predictive ability relies on the quality of the imaging system. In our study, the detailed information of the defects was lost due to exposure. Third, the method did not make use of the successive online images taken per week in discriminating the tiny defects from backgrounds. Tracking and predicting the growth of each defect in successive online images will be an important topic for future research.

References

1. M. L. Spaeth, K. R. Manes, D. H. Kalantar, P. E. Miller, J. E. Heebner, E. S. Bliss, D. R. Speck, T. G. Parham, P. K. Whitman, P. J. Wegner, P. A. Baisden, J. A. Menapace, M. W. Bowers, S. J. Cohen, T. I. Suratwala, J. M. Di Nicola, M. A. Newton, J. J. Adams, J. B. Trenholme, R. G. Finucane, R. E. Bonanno, D. C. Rardin, P. A. Arnold, S. N. Dixit, G. V. Erbert, A. C. Erlandson, J. E. Fair, E. Feigenbaum, W. H. Gourdin, R. A. Hawley, J. Honig, R. K. House, K. S. Jancaitis, K. N. LaFortune, D. W. Larson, B. J. Le Galloudec, J. D. Lindl, B. J. MacGowan, C. D. Marshall, K. P. McCandless, R. W. McCracken, R. C. Montesanti, E. I. Moses, M. C. Nostrand, J. A. Pryatel, V. S. Roberts, S. B. Rodriguez, A. W. Rowe, R. A. Sacks, J. T. Salmon, M. J. Shaw, S. Sommer, C. J. Stolz, G. L. Tietbohl, C. C. Widmayer, and R. Zacharias, *Fusion Sci. Technol.* **69**, 25 (2016).
2. A. Casner, T. Caillaud, S. Darbon, A. Duval, I. Thfouin, J. P. Jadaud, J. P. LeBreton, C. Reverdin, B. Rosse, R. Rosch, N. Blanchot, B. Villette, R. Wrobel, and J. L. Miquel, *High Energy Dens. Phys.* **17**, 2 (2015).
3. Z. He and L. Sun, *Appl. Opt.* **54**, 9823 (2015).
4. G.-H. Hu, Q.-H. Wang, and G.-H. Zhang, *Appl. Opt.* **54**, 2963 (2015).
5. W. Zhu, L. Chen, Y. Liu, Y. Ma, D. Zheng, Z. Han, and J. Li, *Appl. Opt.* **56**, 7435 (2017).
6. F. L. Ravizza, M. C. Nostrand, L. M. Kegelmeyer, R. A. Hawley, and M. A. Johnson, *Proc. SPIE* **7504**, 75041B (2009).
7. L. M. Kegelmeyer, P. W. Fong, S. M. Glenn, and J. A. Liebman, *Proc. SPIE* **6696**, 66962H (2007).
8. G. M. Abdulla, L. M. Kegelmeyer, Z. M. Liao, and W. Carr, *Proc. SPIE* **7842**, 78421D (2010).

9. G. Liu, F. Wei, F. Chen, Z. Peng, and J. Tang, in *Chinese Conference on Pattern Recognition and Computer Vision* (2018), p. 237.
10. F. Wei, F. Chen, B. Liu, Z. Peng, J. Tang, Q. Zhu, D. Hu, Y. Xiang, N. Liu, Z. Sun, and G. Liu, *Opt. Eng.* **57**, 053112 (2018).
11. T. N. Mundhenk, L. M. Kegelmeyer, and S. K. Trummer, *Proc. SPIE* **10338**, 103380H (2017).
12. O. Ronneberger, P. Fischer, and T. Brox, in *International Conference on Medical Image Computing and Computer-assisted Intervention* (2015), p. 234.
13. T. Falk, D. Mai, R. Bensch, Ö. Çiçek, A. Abdulkadir, Y. Marrakchi, A. Böhm, J. Deubner, Z. Jäckel, K. Seiwald, A. Dovzhenko, O. Tietz, C. Dal Bosco, S. Walsh, D. Saltukoglu, T. L. Tay, M. Prinz, K. Palme, M. Simons, I. Diester, T. Brox, and O. Ronneberger, *Nat. Meth.* **16**, 67 (2019).
14. E. M. Christiansen, S. J. Yang, D. M. Ando, A. Javaherian, G. Skibinski, S. Lipnick, E. Mount, A. O'Neil, K. Shah, A. K. Lee, P. Goyal, W. Fedus, R. Poplin, A. Esteva, M. Berndl, L. L. Rubin, P. Nelson, and S. Finkbeiner, *Cell* **173**, 792 (2018).
15. P. Berger and G. Stein, *Mon. Not. R. Astron. Soc.* **482**, 2861 (2019).
16. X. Dong, C. J. Taylor, and T. F. Cootes, in *European Conference on Computer Vision* (2018), p. 398.
17. P. V. C. Hough, in *2nd International Conference on High-Energy Accelerators and Instrumentation* (1959), p. 554.
18. R. Szeliski, *Computer Vision: Algorithms and Applications* (Springer, 2010).
19. R. Brunelli, *Template Matching Techniques in Computer Vision: Theory and Practice* (Wiley, 2009).
20. OpenCV, https://docs.opencv.org/master/de/da9/tutorial_template_matching.html (2019).
21. F. Chollet, <https://github.com/fchollet/keras> (2015).
22. M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mane, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viegas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng, [arXiv:1603.04467](https://arxiv.org/abs/1603.04467) (2016).
23. F. Milletari, N. Navab, and S. A. Ahmadi, in *Fourth International Conference on 3D Vision* (2016), p. 565.