JASA EXPRESS LETTERS

Tracking transients in steelpan strikes using surveillance technology

Cite as: JASA Express Lett. **2**, 023201 (2022); https://doi.org/10.1121/10.0009532
Submitted: 02 December 2021 • Accepted: 17 January 2022 • Published Online: 16 February 2022

Scott H. Hawley, Andrew C. Morrison and Grant S. Morgan

JASA EXPRESS LETTERS ARTICLE asa.scitation.org/journal/jel

Tracking transients in steelpan strikes using surveillance technology

Scott H. Hawley,[1,a)] Andrew C. Morrison,[2,b)] and Grant S. Morgan[1]
[1]Department of Chemistry & Physics, Belmont University, Nashville, Tennessee 37212, USA
[2]Natural Science Department, Joliet Junior College, Joliet, Illinois 60431, USA
scott.hawley@belmont.edu, amorriso@jjc.edu, grant.morgan@pop.belmont.edu

Abstract: This paper presents advancements in tracking features in high-speed videos of Caribbean steelpans illuminated by electronic speckle pattern interferometry, made possible by incorporating robust computer vision libraries for object detection and image segmentation, and cleaning of the training dataset. Besides increasing the accuracy of fringe counts by 10% or more compared to previous work, this paper introduces a segmentation-regression map for the entire drum surface yielding interference fringe counts comparable to those obtained via object detection. Once trained, this model can count fringes for musical instruments not part of the training set, including those with non-elliptical antinode shapes.

[Editor: D Murray Campbell] https://doi.org/10.1121/10.0009532

Received: 2 December 2021 **Accepted:** 17 January 2022 **Published Online:** 16 February 2022

1. Introduction

In the field of musical acoustics, electronic speckle pattern interferometry (ESPI) is a valuable tool. Vibrating plates and membranes in musical instruments such as violins, guitars, drums, and other instruments are often measured and visualized using ESPI.[1,2] Using time-averaged ESPI, small-amplitude measurements are possible. Light and dark fringes, which are lines of constant surface deformation proportional to the wavelength of the laser light, appear in ESPI images. These images illustrate the operating deflection shapes[3] of vibrating surfaces and are comparable to Chladni patterns. However, Chladni patterns are typically associated with standing wave patterns. Images of rapid transient phenomena require greater sophistication, and their interpretation presents a challenging musical acoustics problem.

Recent work by Hawley and Morrison[4] (hereafter "HM2021") showed the use of a deep neural network based object detector[5] to track transient phenomena in Caribbean steelpans illuminated by ESPI and filmed at 15 037 frames per second.[6] Individual video frames were annotated by crowdsourced human volunteers as part of the "Steelpan Vibrations Project" (SVP)[7] in partnership with the Zooniverse.org.[8] Volunteers annotated images by using a web-based graphical user interface to draw ellipses around antinode regions and to enter an integer corresponding to the number of observed interference fringes or "rings," with 11 serving as a maximum value to indicate any ring counts greater than 10. Aggregated annotations from multiple volunteers were used as a dataset to train and evaluate the object detector code "SPNet," which was then used to predict annotations for the remainder of the videos, yielding preliminary physics results. SPNet used a custom-written neural network object detection scheme based on YOLO9000,[9] predicting elliptical antinode regions and a regression-based count of the interference rings appearing in each antinode. When applied to synthetic or "fake" data made by generating images with rings on noisy backgrounds, SPNet scored fairly high on "regression accuracy" scores, e.g., 77% to 95%. When applied to the "real data" (i.e., real ESPI steelpan images paired with aggregated human annotations) the scores were significantly lower, not exceeding 35%. This poor performance on real data were attributed to having noisy or "unclean" annotations, and the "preliminary physics" conclusions were reported without full confidence. Further hindering the SPNet project were software engineering limitations such as the use of old libraries making it difficult to maintain, and it was unable to perform transfer learning,[10] requiring time-consuming training from scratch.

This paper is intended to follow the publication of HM2021 in rapid succession, because while the latter was in press, we were able to improve upon that work in several ways by taking advantage of newer models and software environments, as well as best practices for data-cleaning that involved rapid iteration between model training and human (re-)annotations. The six key results and features of this new paper are as follows. (1) Better scores for both ring-count accuracy and COCO mAP[11] object detection scores than prior work.[12] These were obtained by separating the task of

[a)]ORCID: 0000-0002-5743-6441.
[b)]ORCID: 0000-0003-2676-4087.

detecting antinodes from the counting of rings in cropped sub-images. Bounding boxes (rather than ellipses) were detected around antinodes. These boxes were then used to crop the images. Then the ring counts were obtained for the cropped (sub-)images. (2) The introduction of a "segmentation regression" mapping method for ring counting which produces results in close agreement with the values from the crop-and-count method. (3) Generalization: The ability for the segmentation-regression mapping model trained on steelpan images to predict ring counts for *other musical instruments* in *inference only* (i.e., without training on any images of such other instruments), including instruments with non-elliptical antinode regions. (4) The release of a new dataset for "real data" of annotations of ESPI images of steelpans. (5) This project's methodology for rapid data cleaning by establishing a feedback loop between model predictions and the graphical data-editor software, prioritizing "top loss" examples so that human data-cleaners' efforts could be directed efficiently. (6) The demonstration of advantages in metric scores, algorithmic features, and code maintainability afforded by leveraging up-to-date, well-maintained machine learning (ML) development libraries such as fast.ai[13] and IceVision[14] over the custom-written, 2017-era object detection code SPNet. This allowed for the quick integration of capabilities such as transfer learning, newer optimizers,[15] and run logging.[16] It is hard to overstate the utility of the nbdev[17] development system for "literate programming," allowing code, documentation, and examples all to be written *as one* as Jupyter notebooks[18] and posted for immediate use by collaborators via Google Colaboratory.[19] The code, documentation, and other supplemental materials such as movies are available at the project website (see Ref. 19).

The task of object detection in this study is typically associated with video surveillance technology used to track human beings in CCTV images or roadway features in the view a self-driving car, with the caveat that our systems operate in a post-processing manner and thus need not operate in real time.

2. Data cleaning workflow

The preceding work[4] offered evidence suggesting that inconsistent annotations in the dataset were primarily responsible for poor performance on the metric of "ring count accuracy," a classification-like score that regarded ring counts within ± 0.5 rings of each other as a match. We sought to achieve better metric scores via more intensive data-cleaning, as well as to explore whether that metric of ± 0.5 was perhaps too stringent to meaningfully measure the model's performance and "believability" as a tool for discovering the physical dynamics of steelpan transients. Rather than starting from the earlier "SPNet" dataset,[4] we chose to start fresh from the aggregated annotations of 15 or more volunteers on Zooniverse.[8] These annotations were found to be at least as noisy as the SPNet dataset and needed data cleaning. For this paper, we refer to the volunteers' annotations as the "pre-cleaned" dataset and show its differences from both our final "real" dataset and the SPNet Real dataset.

To maximize the efficiency of the data-cleaning effort, we augmented a graphical ellipse-editor tool developed for HM2021[4] so that it would show predictions of the neural network models in addition to the (human-supplied) annotations. Figure 1 shows a screenshot of the interactive graphical tool used to clean the dataset. Additionally, the images were displayed to users in the order of decreasing value of the loss functions for these models (i.e., we showed "top losses" first) so that the most "noisy" and problematic annotations (i.e., those most inconsistent with the annotations of similar images, such as antinode regions not being marked) could be dealt with first. Periodically the models were retrained on the progressively cleaned data and the model predictions shown in the editor were updated.

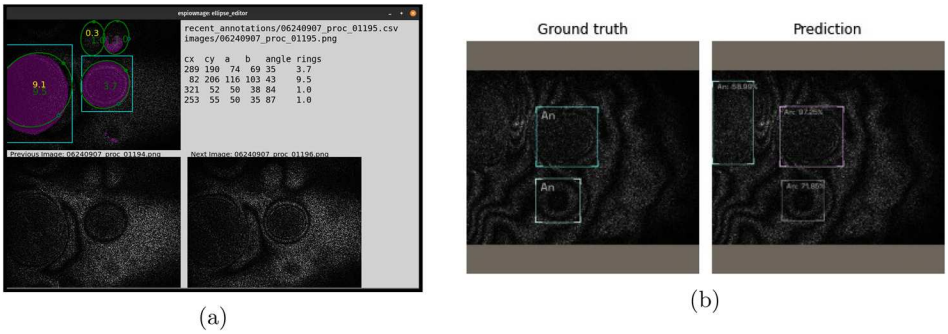

Fig. 1. (a) Screenshot of our enhanced "ellipse editor" tool, which builds on code released in prior work (Ref. 4). Besides the prior ability to graphically edit annotations of boundaries and ring counts elliptical antinode regions, this newer version of the software displays predictions from the neural network models' predictions of bounding boxes, ring counts, and segmentation maps. (b) Bounding box detection of antinode regions via IceVision (Ref. 14) using their tuned RetinaNet (Ref. 20) model. We also tried detecting individual rings as objects but there were too many false negatives, whereas the model was almost always able to detect entire antinodes, including those that human annotators missed. The cropped regions became inputs to the ring-counting code, which is a convolutional neural network that outputs a single logistic regression value where the range has been scaled slightly beyond the maximum number of rings (i.e., we stay within the linear regime of the logistic function). Note that the antinodes are basically circular, becoming more so when cropped and re-shaped as square images, which then allows for arbitrary rotations in addition to other standard image data augmentation methods.

Table 1. Metric scores for different datasets. mAP (mean Average Precision) measures the coincidence of predicted and target bounding boxes (Ref. 11). MAE denotes mean absolute error between the target and predicted ring counts in cropped images. The "$\pm X$" columns show accuracy scores (on [0,1]) for predicted ring counts within a threshold of X rings of their respective target values. Previous results for HM2021 (Ref. 4) were a mAP of 0.64 and accuracy of 0.48 for ± 0.5 rings. Yet ± 0.5 is a tight threshold that may have been unnecessarily stringent; thus in this new work, we include accuracy values for a series of wider thresholds. Note that when $X <$ MAE, accuracy counts tend to be high. Uncertainties of values shown are ± 1 or less in the last digit, except the pre-cleaned dataset for which they are ± 2 in the last digit. Low scores are better for MAE whereas high scores are better for other columns. "New Real" is shown in bold because it comprises the actual physical predictions of the new system.

			Accuracy scores for predicted within $\pm X$ of target				
Dataset	mAP	MAE	± 0.5	± 0.7	± 1	± 1.5	± 2
SPNet Real	0.68	0.85	0.43	0.55	0.70	0.83	0.91
Pre-cleaned	0.66	0.96	0.41	0.52	0.66	0.79	0.87
SPNet CycleGAN	0.73	0.18	0.93	0.96	1.0	1.0	1.0
New Fake	0.865	0.21	0.93	0.97	0.99	1.0	1.0
New Real	**0.68**	**0.71**	**0.53**	**0.66**	**0.79**	**0.89**	**0.94**

3. Models and training

For this work. we dispensed with the end-to-end object-detection-and-ring-count code SPNet and instead opted for a two-stage system consisting of two models: The first was an object detector that only placed bounding boxes around antinode regions. This was done using the pre-trained default model of the object detection system IceVision,[14] trained on our images via Transfer Learning. These bounding boxes would then be used to crop the antinode images, and these cropped images (or "crops") would then be fed into a different model written with `fastai`[13] that use a ResNet[21] convolutional neural network and output a single regression value. Transfer learning and data augmentation were handled essentially automatically by `fastai`. It is noteworthy that the antinode regions are close enough to circular in shape that we could augment the dataset by *arbitrary rotations*.

We developed a new model akin to a segmentation map, however these maps would output a regression value for the entire image, so we termed them "segmentation-regression" maps (although some might regard this as a misnomer, as no actual segmentation is performed). These maps are akin to depth maps and provide some measure of the deformation of the steelpan surface, however they should not be regarded as representing the surface deformation because the segmentation-regression value across an antinode is nearly constant (i.e., it is shaped like a "plateau") with rounded edges [cf. Fig. 4(a)], whereas the physical deformation would be continuously rounded, e.g., ellipsoidal. These segmentation-

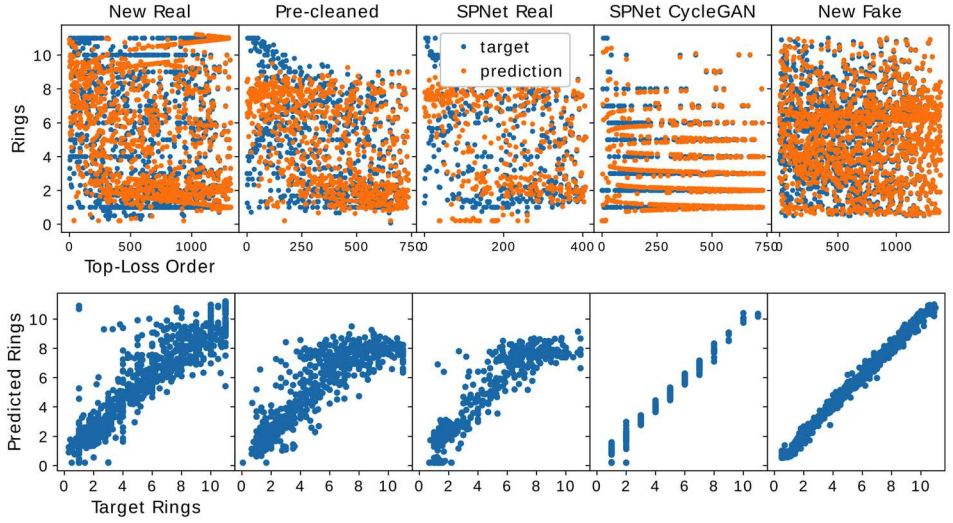

Fig. 2. Ring counts obtained from cropped images. Top row: Predicted and target values, arranged in order of worst agreement to best, for the different datasets. The prevalence of values at the max (11) and min ($\sim$1) are reflections of the data-annotation policy of the SVP (Ref. 7). Note how the CycleGAN dataset from Ref. 4, despite its visual similarity to the real images, only contained integer ring values. Bottom row: Plots of predicted rings vs target rings as a kind of "transfer function" (where a 1:1 linear relationship is "good"), showing that our data-cleaning effort ("New Real," left column) resulted in closer agreement and less compression of the dynamic range that other datasets. The difference between "New Real" and "Pre-cleaned" indicates the improvement obtained by the data-cleaning effort.

Table 2. Akin to Table 1 but for segmentation-regression maps, for mask quantization using a bin size of 0.7, which was the minimum resolution at which the model was able to train effectively. Uncertainties are the same as in Table 1. For CycleGAN data, training stagnated immediately; the reason for this is still under investigation. The scores in this table are not as favorable as those in Table 1. We speculate that this is because these numbers are the result of integrating over the entire image, whereas values sampled from the middle of antinode regions display close agreement with the method of counting rings in cropped images, as demonstrated in Fig. 3(c).

		Accuracy scores for predicted within $\pm X$ of target				
Dataset	MAE	± 0.5	± 0.7	± 1	± 1.5	± 2
SPNet Real	0.61	0.19	0.26	0.36	0.50	0.62
Pre-cleaned	0.74	0.15	0.21	0.29	0.41	0.52
SPNet CycleGAN	0.20	0.00	0.00	0.00	1.00	1.00
New Fake	0.19	0.52	0.66	0.80	0.90	0.94
New Real	**0.75**	**0.19**	**0.25**	**0.37**	**0.49**	**0.62**

regression maps were found to provide a useful alternative model that agreed with the ring counts obtained from cropped images, yet offered some additional flexibility.

4. Results

We show example images of predicted bounding boxes and sample cropped rings in Fig. 1(b). The COCO mAP[11] object detection scores for the bounding boxes and the "regression accuracy" scores for various ring count tolerances are provided in Table 1. These scores are obtained using the "test" subsets which comprise 20% of each of the corresponding five datasets of cropped antinode images, from which the test subsets were withheld during training. Figure 2 illustrates the accuracy of the ring counts for the test subsets of models trained on the full datasets, grouped as columns. The first three columns involve largely the same input images yet have different annotations, demonstrating the effects of the data-cleaning effort of this present paper. The other two columns are synthetic or "fake" data, one for the CycleGAN-processed data of the SPNet Dataset (which only had integer ring counts), the other for a new set of fake data for which ring counts could take on integer values. (Space limitations do not allow us to show these new images but they are available in Ref. 19.)

Similar to Table 1, Table 2 shows metric scores obtained using the segmentation-regression map method. Because the segmentation U-Net[22] model provided by fastai[13] required integer-valued pixels instead of floats, we quantized the ring counts at the minimum-trainable resolution of 0.7 rings. (Below that resolution, the model would not train.) Figure 3 demonstrates the efficacy of our methods, and their improvements on prior predictions. Noteworthy is our confirmation of "preliminary physics" results in HM2021,[4] namely, that there is a delay in the rise of the vibration amplitude of the octave note's fundamental resonance which is large compared to the time for vibrations to move through the steelpan. Additionally, Fig. 3 shows the octave note's amplitude measurements given by our improved method can be more closely fit to a function corresponding to the frequency of oscillation of the octave note. The inclusion of Fig. 3(b) is intended to show further confirmation of the measurements in HM2021, which included three other steelpan strikes. One interpretation of Fig. 3(b) is that the majority of the audio production of the second harmonic tone is due to the excitation of the struck note. For the other strikes in HM2021 the interpretations may not be as straightforward, and await further

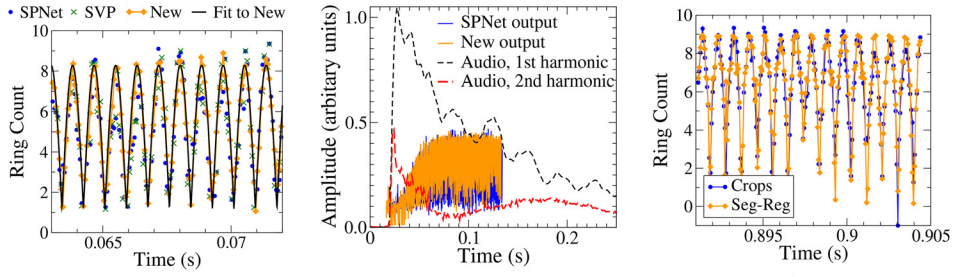

Fig. 3. (a) Our new version of Fig. 6 from HM2021 (Ref. 4), in which the fit is given by $y = A|\cos(Bt + C)| + D$, with $A = 7.11 \pm 0.080$, $B = 3748 \pm 0.68$, $C = -0.4470 \pm 0.05$, and $D = 1.17 \pm 0.057$. These constitute lower uncertainties than the prior work, furthermore the RMSE value of 0.48 and correlation coefficient of 0.97 indicate a closer fit than in the prior work. (b) Our replication of a key "preliminary physics" result in a panel from Fig. 7 of HM2021 (Ref. 4), showing a delay in rise time of the amplitude of the lowest resonance of the octave note as observed in ESPI video as compared to the amplitudes of the first and second harmonics from the audio recording (dot-dashed/black and dashed/red lines). (c) Close agreement between the ring counts (as a function of time) from the bounding-box-and-crop method and the segmentation-regression method, when the latter is measured from the center of an antinode region.

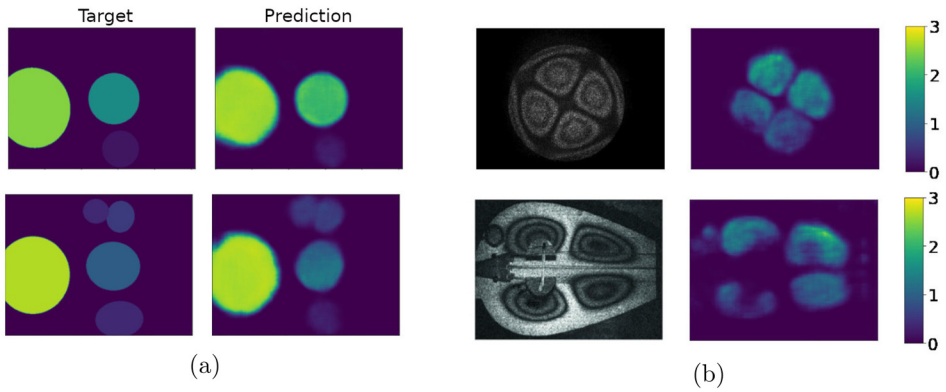

Fig. 4. (a) Comparison of target and predicted segmentation-regression maps for steelpan images, showing that the edges of the antinode regions tend to be rounded in the predictions. (b) Segmentation-regression maps for other instruments, highlighting the generalization performance of our model. These maps were obtained via inference using the model trained only on our steelpan dataset. Although the model was only trained on elliptical antinode regions, we see that it is able to trace the contours and count rings for the roughly triangular antinode regions for the drum image on the top (Ref. 23) and the bean-shaped antinodes of the 17th-century lyra on the bottom (Ref. 2).

analysis. Figure 4 shows examples of the segmentation-regression mapping. Notably Fig. 4(b) shows that the model is able to predict ring counts for images far outside the training distribution, namely, instruments not included in the training dataset having non-elliptical antinode shapes: a different type of drum and a 17th century lyra. The generalization performance of the model is still under investigation. Due to space limitations, further examples, including images for which the model fails to make a prediction, can be found in Ref. 19.

5. Conclusions

The methods used in this paper confirm prior work and improve upon it in the form of tighter error bars, higher accuracy and object detection scores, and generalization to other musical instruments. In particular, the segmentation-regression mapping method yields measurements comparable to counting rings in cropped images. We now have segmentation-regression values for all pixels of all video frames and have enabled a feature in the ellipse editor such that a user can right-click on any pixel and immediately obtain a time-series graph of ring-count values as a function of time at that location. We expect this will be a valuable tool for exploring the physics of the drum surface oscillations. We note, however, that ESPI images are limited by an interpretive difficulty when trying to discern higher harmonic excitation of notes, and indeed the Steelpan Vibrations Project[7] annotation process had no provision for labeling higher harmonics. Laser Doppler vibrometry would be more suitable for discerning, for example, the relative contributions of the struck note and octave note oscillations to the audio recorded in Fig. 3(b).

Though this project was computation-intensive, the total power consumption was ∼15 kWh, indicating a minimal environmental impact. The training was performed on two personal workstations with a total of 4 NVIDIA GPUs (RTX 3080, 2080Ti, and two Titan X's), though all computations are reproducible on Google Colab via our provided code-and-documentation Jupyter notebooks hosted with other supplemental materials in Ref. 19.

Acknowledgments

Andrew Morrison receives support through NSF Grant No. 1741934. The authors thank Zach Mueller for his assistance with `fastai` and nbdev-based development, and to Farid Hassainia for his help with IceVision. This publication uses data generated via the Zooniverse.org platform, development of which is funded by generous support, including a Global Impact Award from Google, and by a grant from the Alfred P. Sloan Foundation.

References and links

[1]T. Moore, "Measurement techniques," in *Springer Handbook of Systematic Musicology* (Springer, Berlin, 2018), pp. 81–103.
[2]E. Bakarezos, Y. Orphanos, E. Kaselouris, V. Dimitriou, M. Tatarakis, and N. A. Papadogiannis, "Laser-based interferometric techniques for the study of musical instruments," in *Current Research in Systematic Musicology* (Springer International, New York, 2019), pp. 251–268.
[3]M. H. Richardson, "Is it a mode shape, or an operating deflection shape?," Sound Vib. **31**, 54–61 (1997) .
[4]S. H. Hawley and A. C. Morrison, "Convnets for counting: Object detection of transient phenomena in steelpan drums," J. Acoust. Soc. Am. **150**(4), 2434–2445 (2021).
[5]Z.-Q. Zhao, P. Zheng, S.-T. Xu, and X. Wu, "Object detection with deep learning: A review," IEEE Trans. Neural Netw. Learn. Syst. **30**(11), 3212–3232 (2019).
[6]A. C. Morrison, T. R. Moore, and D. Zietlow, "High speed electronic speckle pattern interferometry as a method for studying the strike on a steelpan," J. Acoust. Soc. Am. **129**(4), 2615–2615 (2011).

[7]A. C. Morrison, "Steelpan Vibrations," Zooniverse.org (2017) https://www.zooniverse.org/projects/achmorrison/steelpan-vibrations (Last viewed 2/10/2022).
[8]K. D. Borne and Z. Team, "The Zooniverse: A Framework for Knowledge Discovery from Citizen Science Data," in *AGU Fall Meeting Abstracts* (2011).
[9]J. Redmon and A. Farhadi, "YOLO9000: Better, faster, stronger," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, Honolulu, HI (2017), pp. 6517–6525.
[10]K. Wang, X. Gao, Y. Zhao, X. Li, D. Dou, and C.-Z. Xu, "Pay attention to features, transfer learn faster CNNs," in *International Conference on Learning Representations* (2020), https://openreview.net/forum?id=ryxyCeHtPB (Last viewed 2/10/2022).
[11]T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *Computer Vision—ECCV 2014*, edited by D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars (Springer International Publishing, Cham, 2014), pp. 740–755.
[12]MAP (mean average precision) is a standard object detection metric originating with MS-COCO (Ref. 11). mAP averages the intersection-over-union areas of predicted and target bounding boxes over a series of detection thresholds.
[13]J. Howard and others, "fastai" (2018) https://github.com/fastai/fastai (Last viewed 2/10/2022).
[14]L. Vazquez and F. Hassainia, "Icevision: An agnostic computer vision framework" (2020), https://github.com/airctic/IceVision (Last viewed 2/10/2022).
[15]L. Wright, "Ranger—A synergistic optimizer" (2019), https://github.com/lessw2020/Ranger-Deep-Learning-Optimizer (Last viewed 2/10/2022).
[16]L. Biewald, "Experiment tracking with weights and biases" (2020), software available from wandb.com, https://www.wandb.com/ (Last viewed 2/10/2022).
[17]J. Howard and others, "nbdev" (2019) https://github.com/fastai/nbdev (Last viewed 2/10/2022).
[18]T. Kluyver, B. Ragan-Kelley, F. Pérez, B. Granger, M. Bussonnier, J. Frederic, K. Kelley, J. Hamrick, J. Grout, S. Corlay, P. Ivanov, D. Avila, S. Abdalla, and C. Willing, "Jupyter notebooks—A publishing format for reproducible computational workflows," in *Positioning and Power in Academic Publishing: Players, Agents and Agendas*, edited by F. Loizides and B. Schmidt (IOS Press, Amsterdam, 2016), pp. 87–90.
[19]For reproducibility, documentation of results in the form of executable Colab notebooks and supplemental materials such as movies are hosted at https://drscotthawley.github.io/espiownage/ (Last viewed 2/10/2022).
[20]T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *2017 IEEE International Conference on Computer Vision (ICCV)* (2017), pp. 2999–3007.
[21]K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2016), pp. 770–778.
[22]O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Medical Image Computing and Computer-Assisted Intervention—MICCAI 2015*, edited by N. Navab, J. Hornegger, W. M. Wells, and A. F. Frangi (Springer International Publishing, Cham, 2015), pp. 234–241.
[23]T. R. Moore, "A simple design for an electronic speckle pattern interferometer," Am. J. Phys. **72**(11), 1380–1384 (2004).