A Real-Time Machine Learning Based Solution for Privacy Enforcement in Video Recordings and Live Streaming

Pietro Manganelli Conforti[1], Matteo Emanuele[1] and Lorenzo Mandelli[1]

[1]Department of Computer, Control and Management Engineering, Sapienza University of Rome, Italy

Abstract

These past years the world had to deal with a whole new situation brought by Covid-19. Everyone's routine changed and we started passing way more time than before on virtual meeting, virtual chats and similar. With this, many privacy problems arised from all the video data generated by a single user. Google and Zoom introduced the possibility to blur out the background while using a front face camera, but this did not solve many privacy concerns ranging from showing people in videos without their permission, to the leaking of sensible data and information from videos uploaded online. We propose a solution build over the use of computer vision techniques like image segmentation and classification for context recognition for a privacy enforcement solution capable of fitting the user's personal need, blurring out selectively specific objects from a video based on the user's preferences for each room in which they are.

Keywords

Image segmentation, Context Recognition, Detectron2, Privacy enforcement, Covid-19, Alexnet, Transfer learning,

1. Introduction

In the past years there has been a solid shift for the entire world population towards a more active presence online. Covid-19 has further pushed many activities to be faced digitally. Virtual meeting application like Zoom, had 10 million daily meeting participants in December 2019, but by April 2020, that number increased to reach up to 300 million [1]. It is estimated that in 2024 only 25% of the business meetings will take place in person [2]. Studies started during 2020 have demonstrated that nowadays people spend on average way more time in virtual meetings than before [3], leading to many concerns for the single person. Users have started experiencing stress related to not being competent in the use of the technology, but most importantly to "Zoom fatigue" due to it being always "on" [4]. Many privacy related issues have been crippling the user experience ever since, such as exposing private and personal spaces on camera, unintentionally framing a person who did not give consent to be on video or sharing sensible information leaked from careless online posting. Many solutions have been promptly developed to prevent such things from happening, providing virtual meeting room services with safeguard-privacy functionalities like blurring the background and virtual backgrounds[5, 6, 7, 8]. We present in this paper a novel computer vision based approach for privacy enforcement in video data, capable of filter out from a video a list of objects that a user does not want to show, based on the recognition of the environment framed, in order to blur out objects relatively to both the user needs and the context in which they are.

2. Related works

With the advancement of technology, people have been sharing a continuously growing amount of personal data online. Additionally to life-logging devices[9], social medias have recently stepped in, ending up quickly dominating the landscape of mass produced data with "visual data"(i.e. images and video). For instance, in 2020, the first year of pandemic, users have generated and shared via Facebook a total of 10.5million videos[10].
This impactful amount of data brought to the attention of experts and users to many privacy related issues; studies started identifying and observing how easily privacy could be violated just from unintentionally sharing personal data contained inside images and videos, and subsequently started proposing privacy models to formally approach and tackle said scenarios [11]. The scientific world went quickly from defining sub-fields like Privacy-Preserving Machine Learning(PPML) [12], to adopting deep learning models for image disguising [13], Context Recognition[14] as well as Image-Based Localization[15] and again computer vision based framework[16] as novel solutions for privacy preserving of first person vision image sequences, placing computer vision, Artificial Intelligence and data driven approaches as state of the art techniques for preserving privacy on line. Among the many available solutions for privacy preservation and safeguarding, it is currently missing one which allows single users to selectively censor objects from visual

ICYRIME 2024: 9th International Conference of Yearly Reports on Informatics, Mathematics, and Engineering. Catania, July 29-August 1, 2024

mandelli@diag.uniroma1.it (L. Mandelli)

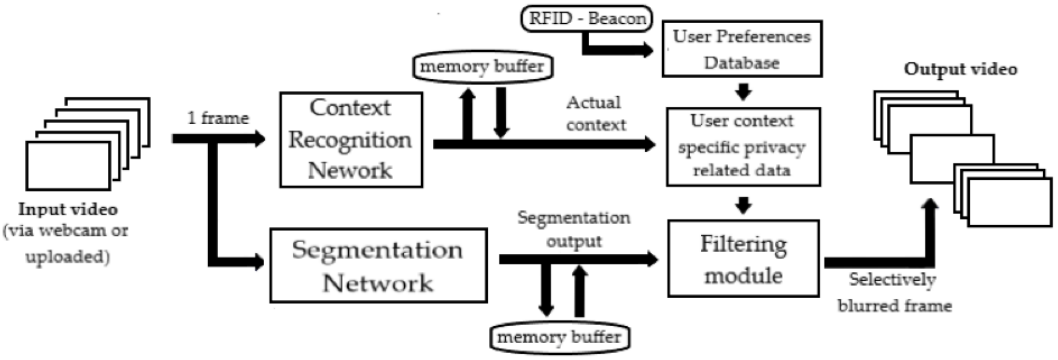

Figure 1: The pipeline of our system.

data depending on their personal needs and preferences. The propose work aims therefore to provide experience-lacking users with an intuitive, easy to use tool for privacy enforcement in video data based on computer vision techniques.

3. Implementation

Sensible users' data is crucial to be kept private. The proposed software tracks such information by means of various modules which pipeline is shown in fig.1. Memory buffers are used in between modules to guarantee flexibility towards input videos of any $aspect$, $ratio$ and fps, as well as to stabilize the output overcoming the common flickering experienced in these kind of applications. Thanks to such buffers it is possible to store past frames and reuse those to statistically smooth of the final output; past frames are reused according to level of confidence the class recognition module predicted with. Input videos can be directly uploaded to the system or streamed from cameras(i.e. webcams).

The proposed solution separates the overarching learning problem in two sub problems, namely context recognition and image segmentation; this approach guarantees robustness through modularity and simplifies the overall functioning of the software.

Recognizing the users, their emotional state [17, 18, 19], his attentive state [20, 21, 22], and the context surroundings[23, 24, 25] allows to selectively obscure specific elements based on preferences the same user expressed at the registration time; such data is stored into a database for later inference. Context recognition has been tackled with a neural network inspired by Alexnet [26], a famous Deep Convolutional Neural Network, designed for image classification.

Together with an RFID application [14] Detectron 2, a very powerful instance segmentation network published by Facebook in 2019 [27], is used to identify user's context specific, privacy related data, within video frames, with no ambiguity; combining the output masks produced by Detectron 2 with all the information retrieved before, a particular region of the frame is identified and filtered with a Gaussian transform. Similar or identical contexts disambiguation is possible to be tackled and solved with the support of RFID technology: with the introduction of a beacon that send a constant signal, it is possible to recognize and distinguish two apparently-same looking environment. Such discriminatory action is essential, yet simple to be applied since it is integrable in any environment with low effort or invasiveness. A similar RFID-based solution for context recognition was already presented by another research [14] conducted some years ago. Finally the desired effect is obtained by processing and collecting all the frames of the video and setting the right frame velocity.

3.1. Dataset

Distinct datasets have been used for the two different learning tasks respectively, image segmentation and context recognition. The choice of using the Detectron2 network for the the image segmentation tasks leaves little to no choice but using the 2017 version for the COCO dataset [28] which has been demonstrated to be performative with such dataset.

COCO is a dataset composed of two groups of elements: images and annotations. Images contain a vast variety of objects, for a total of 80 different category of elements. The network was capable to recognize them all, and even apparently odd objects were left untouched and not removed. Together with the set of images, COCO is composed of a set of so-called "annotations" that contain information related to the position of the object masks,

their bounding box and their location on the image reference frame.
For what concern the context recognition part, a slightly modified version of a dataset available on Kaggle[29] has been used; such dataset is composed of 5 different classes symbolizing five different kind of rooms, two of which has been merged together, namely living room and dining room. Each element is originally an RGB picture of a fixed size of 224x224x3 which has been resized to be 227x227x3, to better fit through AlexNet (?).
As part of the training & testing process a defined set of image processing techniques have been organized into a pipeline. Such transformation pipeline has been implemented using Albumentation library[30], an easy-to-use and intuitive library for image processing; it consists of: *ShiftScaleRotate*, for shifting or rotating images; *RGBshift* for randomly altering RGB channels' values; *RandomBrightnessContrast* for randomly changing iamges' brightness and contrast; *MultiplicativeNoise* for randomly adding noise; *Normalize*, for normalizing data; *HueSaturationValue*, for randomly changing images' saturation.

3.2. Image Segmentation Network

Detectron2 is Facebook AI Research's library [27] that provides state-of-the-art detection and segmentation algorithms. It is the successor of Detectron [31] which is in turn based on the Maskrcnn-benchmark model [32]. It supports a great number of computer vision research projects thanks to its flexibility, output capabilities and available documentation.
Among the available Detectron2' architectures mask-rcnn-fpn has been chosen. Such architecture is mainly built from three modules: a Backbone Network, a Region Proposal Network and a Box Head.
The $Backbone\ Network$, whose role is to extract multi-scale feature maps with different receptive fields starting from the input image, is based on the Feature Pyramid Network [33] technique. In this way areas of interest from different points of view are identified and passed to both the two next modules. The $Region\ Proposal\ Network$ detects object regions (which are the so called "proposal boxes") based on multi-scale features, which together with the feature maps serve as input for Roi (Region of interest) $Head$. This last module warps feature maps using proposal boxes into multiple fixed-size features, and retrieves the fine-tuned box locations and classification results via fully-connected layers.

3.3. Context Recognition Network

The context recognition task belongs to a classification problem, where each frame of the video is treated as an image to classify. For this task Alexnet has been fine tuned on our reduced dataset by means of transfer learning.
The structure of the network is shown in figure 2.

Here we report the performance with which we evaluated our model. We will give particular importance to both the accuracy and F1-score of each class.

Classes	precision	recall	F1-score	overall Accuracy
Bathroom	0.84	0.90	0.87	
LivingRoom	0.92	0.79	0.85	
Bedroom	0.69	0.76	0.76	
Kitchen	0.75	0.76	0.76	
				0.83

Table 1
Performance of our classification task

The performance of the network are reported in the table 1. As we can see we are capable of obtaining high F1-score values for each class and an overall accuracy above 80%, making the results satisfying for our standards. We can consider the macro F1 score as a general metric of evaluation, defined as:

$$macro - F1 = \frac{1}{N}\sum_{i=1}^{N} F1_i = 0.81$$

Which is simply an overall F1 score among all classes. In our case, we can see a macro F1 score of 0.81.

It is also possible to take vision of the confusion matrix in figure 3, showing how the different samples from the test set were classified during the test phase.

3.4. Output stabilization techniques

Videos in daily life scenarios, likely happen to contain temporary blank frames, as well as artifacts, due to user or scene related conditions. A context recognition network will therefore generate a classification label that will be trivially assigned, leading to an instability problem. We dealt with this problem through the introduction of a "memory buffer", that is capable of stabilizing statistically the result.

This is achieved by endowing the system with two buffers, one for the context recognition network that stores the predicted context classes, and one used to track the instance segmentation network output and to store the classes predicted with enough accuracy. Storage of past data allows to create a time relation between successive frames, thus enforcing the output of each network and stabilizing the final one. This method allows to correlate information inside the video with the least expenditure of resources. There are two buffers.

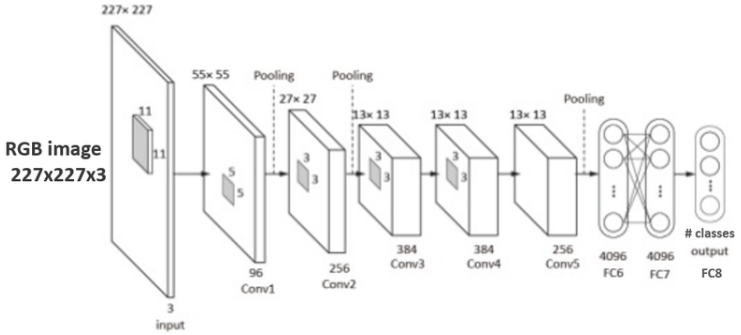

Figure 2: Alexnet architecture. All rights reserved to the owner of the picture[34]

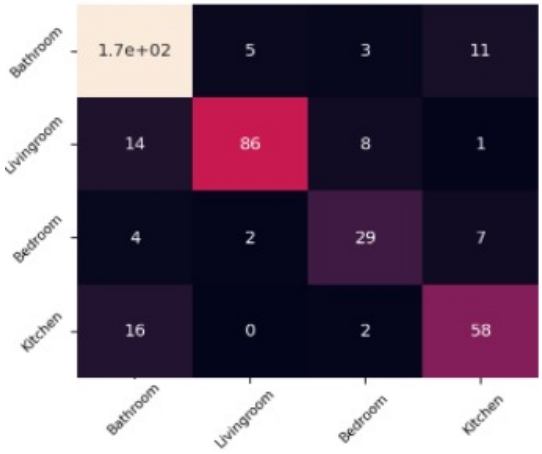

Figure 3: The confusion matrix generated with the use of the scikit library [35]

3.4.1. Context memory buffer

The first buffer visible in the top of fig.1 is the one dedicated to the output of Alexnet. The assumption taken by this approach is that the context changes don't take place suddenly but instead are related to a smooth trend. For instance, if the frame n recognizes a specific context, there is a high probability that also the frame $n + 1$ will carry out similar information and represents the same context. Thus, doing an average among the past frames increases the overall accuracy by smoothing the output trend.
The length of the buffer is set dynamically in relation to the fps value retrieved from the video and the information obtained by the frames from the last half of second gets stored within.
The trade-off of this method is a small delay from the context recognition module because the output context label has to stabilize for at least half of the buffer $length$ to change the output and in this short period of time is wrongly classified with the previous stable label. This is strongly compensated by the stability provided and the delay is short enough to be difficult to notice.

3.4.2. Instance segmentation memory buffer

The other output stabilization technique is the instance segmentation memory buffer dedicated to the Detectron output. The rationale here is regarding the threshold used by the model to decide if an element belongs or not to a certain class. Being a privacy concern to conceal as much as possible the sensible information inside the frames, false positives in exchange of a higher number of true positives are preferable. Therefore, two kinds of thresholds are considered: the basic one and the optimal one. The first one is lower than the second one and is the minimal value considered acceptable to take into account the output of the network. If the output confidence regarding a specific instance inside the frame falls below this value is considered too unclear and it is not counted. The second threshold instead represents the optimal value of confidence used by the system in order to properly recognize an element with enough accuracy. This information is used in order to track the last elements appeared to the network, appending them inside the buffer.
If the networks find an instance of a class inside the buffer even with a lower confidence value in respect to the optimal threshold it is still considered acceptable, therefore processed and eventually concealed. In this way it gets easier for the network to work with moving objects because this method allows it to trace them even in the case of uncertainty due to movement.
The buffer length is dynamically related to the fps value and stores information regarding the set of frames concerning the last three seconds. If an element is recognized by the network after this time interval in order to be evaluated it needs to overcome the optimal confidence threshold again.

Figure 4: a: kitchen **b:** Bathroom **c:** Bedroom

The trade-off of this system, as mentioned above, is a higher frequency of false positives, which can be misleading for the final result and inversely proportional in number to the two threshold values. Overall, the accuracy following this approach improved by a discrete percentage, mainly in the more dynamic scenarios.

4. Results

The results of the system are evaluated accordingly on how many times the full procedure works consistently to specific information given as input related to a specific test video. Knowing in advance those information which are the settings inside a set of test videos as well as the list of elements inside of them, we can measure the overall accuracy of the system.
For instance, if a video displays a specific context with a certain number of known elements inside of it we can control how many times those elements are found by the two networks and by the two memory buffers in the output trend.
In order to achieve this result an evaluation procedure was implemented that given an input video follows similar steps that the system does but keeps into account the number of times the output given starting from each input frame is correct in respect to the total number of them. The test video used are three, which are providing the following scenarios:

- $kitchen$ (fig. 4.a), where we want to blur out a bowl from a table given that the system recognize the context. The instance segmentation network can identify various objects as a table, a oven and bottles. Due to the user preferences, here the stationary objects we want to blur out from all the video frames is simply one, a black bowl.

- $bathroom$ (fig. 4.b), where the user want to blur out the WC. In such scenario, the instance segmentation network recognize as a WC also the bidet given the similarity in their structure.

- $Bedroom$ (fig. 4.c), In this scenario there is a bowl placed in a flat surface behind the bed, and the user wants to blur it out. This scenario can be potentially challenging since the portion of the room we are framing is very restricted, and the only object that can be considered a strong feature is the bed.

The results are shown in the following table. Here we have 5 different values of evaluation for each test:

- *accuracy of context recognition network respect the total number of frames(C.R.)*, indicating the percentage of success for the context recognition network applied to the frames of the video. This value does not show the improvements brought by the memory buffer.

- *accuracy of context recognition network + memory buffer respect the total number of frames (C.R. $\backslash w$ B.)*, indicating the percentage of success for the context recognition network combined with the memory buffer for the context recognition task.

- *Accuracy of the instance segmentation network in finding the objects of interest in a frame respect the total number of frame(I.S.).* This indicates the percentage of success for the instance segmentation task respect the objects of interest for the user. This value does not show the improvements brought by the memory buffer.

- *Accuracy of the instance segmentation network + memory buffer in finding the objects of interest in a frame respect the total number of frame(I.S. $\backslash w$ B.).* this indicates the percentage of success for the instance segmentation task respect the objects of

interest for the user. This value does not show the improvements brought by the memory buffer.

- *overall accuracy of the whole system.* This indicates, as the name states, the overall accuracy of the whole pipeline. This accuracy is given as a combined accuracy from the accuracy of the two tasks, obtained as the product between the accuracy of the instance segmentation considering also the buffer and the context recognition considering also the buffer.

In table 2 tests' results are reported, confirming that memory buffers contribute to increase the accuracy of both tasks, translating in overall better system's accuracy.
It must be noted that accuracy can be further improved by fine tuning the thresholds required by the instance segmentation task. A general thumb rule is that, if the accuracy is similar between the system using the buffers and the system not using it, it is possible to improve the performance through such fine tuning.

Input	C.R.	C.R. \w B.	I.S.	I.S. \w B.	Overall %
Kitchen	0.91	1.0	0.89	0.98	0.98
Bathroom	0.74	0.81	0.89	1.0	0.81
Bedroom	0.92	1.0	0.5	0.78	0.78

Table 2
Performance of the system

5. Conclusions

In this paper we presented a machine learning-powered solution for privacy enforcement in video data, a data-driven implementation to safeguard the privacy of any user that may be forced to spend plenty of hours in videos and/or in video meetings. Such solution will solve an untouched problem that wasn't formally faced by the field in the past years, where our time started to be off-centre towards the time spent online.
Our implementation presents good performance even in presence of noisy or foggy videos, while perform almost perfectly in the most common scenarios of videos with common perspectives extracted from general recordings using mobile devices. The adaptability of the system to change with the needs of different users both for the objects of interest and the context of interest makes the solution we propose a solid step forward in the field of privacy enforcement for video data.

References

[1] B. Evans, The zoom revolution: 10 eye-popping stats from tech's new superstar, 2020.
[2] W. Standaert, S. Muylle, A. Basu, How shall we meet? understanding the importance of meeting mode capabilities for different meeting objectives, Information Management (2021). doi:https://doi.org/10.1016/j.im.2020.103393.
[3] D. Chew, M. Azizi, The state of video conferencing 2022, 2022.
[4] K. A. Karl, J. V. Peluchette, N. Aghakhani, Virtual work meetings during the covid-19 pandemic: The good, bad, and ugly, Small Group Research 0 (2021). URL: https://doi.org/10.1177/10464964211015286.
[5] M. Wozniak, C. Napoli, E. Tramontana, G. Capizzi, G. Lo Sciuto, R. K. Nowicki, J. T. Starczewski, A multiscale image compressor with rbfnn and discrete wavelet decomposition, in: Proceedings of the International Joint Conference on Neural Networks, volume 2015-September, 2015. doi:10.1109/IJCNN.2015.7280461.
[6] G. Capizzi, S. Coco, G. L. Sciuto, C. Napoli, A new iterative fir filter design approach using a gaussian approximation, IEEE Signal Processing Letters 25 (2018) 1615 – 1619. doi:10.1109/LSP.2018.2866926.
[7] D. Połap, M. Woźniak, C. Napoli, E. Tramontana, R. Damaševičius, Is the colony of ants able to recognize graphic objects?, Communications in Computer and Information Science 538 (2015) 376 – 387. doi:10.1007/978-3-319-24770-0_33.
[8] M. Woźniak, D. Połap, M. Gabryel, R. K. Nowicki, C. Napoli, E. Tramontana, Can we process 2d images using artificial bee colony?, in: Lecture Notes in Artificial Intelligence (Subseries of Lecture Notes in Computer Science), volume 9119, 2015, p. 660 – 671. doi:10.1007/978-3-319-19324-3_59.
[9] A. L. Allen, Dredging up the past: Lifelogging, memory and surveillance, University of Chicago Law Review 12 (2008) 2825–2830.
[10] T. Dobrilova, The most astonishing facebook statistics in 2022, 2022. URL: https://techjury.net/blog/facebook-statistics/.
[11] S. Cunningham, M. Masoodian, A. Adams, Privacy issues for online personal photograph collections, Journal of Theoretical and Applied Electronic Commerce Research 5 (2010). doi:10.4067/S0718-18762010000200003.
[12] R. Xu, N. Baracaldo, J. Joshi, Privacy-preserving machine learning: Methods, challenges and directions, 2021. arXiv:2108.04417.
[13] S. Sharma, K. Chen, Image disguising for privacy-preserving deep learning, in: Proceedings of the 2018 ACM SIGSAC Conference on Computer and

Communications Security, Association for Computing Machinery, New York, NY, USA, 2018. URL: https://doi.org/10.1145/3243734.3278511.

[14] G. M. Farinella, C. Napoli, G. Nicotra, S. Riccobene, A context-driven privacy enforcement system for autonomous media capture devices, Multimedia Tools and Applications 78 (2019) 14091–14108. URL: https://doi.org/10.1007/s11042-019-7376-z.

[15] P. Speciale, J. L. Schönberger, S. B. Kang, S. N. Sinha, M. Pollefeys, Privacy preserving image-based localization, CoRR abs/1903.05572 (2019).

[16] A. T.-Y. Chen, M. Biglari-Abhari, K. I.-K. Wang, Trusting the computer in computer vision: A privacy-affirming framework, in: 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2017.

[17] I. E. Tibermacine, A. Tibermacine, W. Guettala, C. Napoli, S. Russo, Enhancing sentiment analysis on seed-iv dataset with vision transformers: A comparative study, in: ACM International Conference Proceeding Series, 2023, p. 238 – 246. doi:10.1145/3638985.3639024.

[18] V. Ponzi, S. Russo, A. Wajda, R. Brociek, C. Napoli, Analysis pre and post covid-19 pandemic rorschach test data of using em algorithms and gmm models, in: CEUR Workshop Proceedings, volume 3360, 2022, p. 55 – 63.

[19] A. Alfarano, G. De Magistris, L. Mongelli, S. Russo, J. Starczewski, C. Napoli, A novel convmixer transformer based architecture for violent behavior detection, in: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), volume 14126 LNAI, 2023, p. 3 – 16. doi:10.1007/978-3-031-42508-0_1.

[20] E. Iacobelli, V. Ponzi, S. Russo, C. Napoli, Eye-tracking system with low-end hardware: Development and evaluation, Information (Switzerland) 14 (2023). doi:10.3390/info14120644.

[21] F. Fiani, S. Russo, C. Napoli, An advanced solution based on machine learning for remote emdr therapy, Technologies 11 (2023). doi:10.3390/technologies11060172.

[22] V. Ponzi, S. Russo, V. Bianco, C. Napoli, A. Wajda, Psychoeducative social robots for an healthier lifestyle using artificial intelligence: a case-study, in: CEUR Workshop Proceedings, volume 3118, 2021, p. 26 – 33.

[23] R. Brociek, G. D. Magistris, F. Cardia, F. Coppa, S. Russo, Contagion prevention of covid-19 by means of touch detection for retail stores, in: CEUR Workshop Proceedings, volume 3092, 2021, p. 89 – 94.

[24] S. Pepe, S. Tedeschi, N. Brandizzi, S. Russo, L. Iocchi, C. Napoli, Human attention assessment using a machine learning approach with gan-based data augmentation technique trained using a custom dataset, OBM Neurobiology 6 (2022). doi:10.21926/obm.neurobiol.2204139.

[25] E. Iacobelli, S. Russo, C. Napoli, A machine learning based real-time application for engagement detection, in: CEUR Workshop Proceedings, volume 3695, 2023, p. 75 – 84.

[26] A. Krizhevsky, I. Sutskever, G. E. Hinton, Imagenet classification with deep convolutional neural networks (2012).

[27] Y. Wu, A. Kirillov, F. Massa, W.-Y. Lo, R. Girshick, Detectron2, https://github.com/facebookresearch/detectron2, 2019.

[28] T.-Y. Lin, M. Maire, S. Belongie, L. Bourdev, R. Girshick, J. Hays, P. Perona, D. Ramanan, C. L. Zitnick, P. Dollár, Microsoft coco: Common objects in context, 2015. arXiv:1405.0312.

[29] RobinReni, House rooms image dataset, 2020. URL: https://www.kaggle.com/robinreni/house-rooms-image-dataset.

[30] A. Buslaev, V. I. Iglovikov, E. Khvedchenya, A. Parinov, M. Druzhinin, A. A. Kalinin, Albumentations: Fast and flexible image augmentations, Information 11 (2020) 125. URL: http://dx.doi.org/10.3390/info11020125.

[31] R. Girshick, I. Radosavovic, G. Gkioxari, P. Dollár, K. He, Detectron, 2018. URL: https://github.com/facebookresearch/detectron.

[32] F. Massa, R. Girshick, maskrcnn-benchmark: Fast, modular reference implementation of Instance Segmentation and Object Detection algorithms in PyTorch, https://github.com/facebookresearch/maskrcnn-benchmark, 2018. Accessed: [Insert date here].

[33] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, S. Belongie, Feature pyramid networks for object detection, 2017. arXiv:1612.03144.

[34] A. Khvostikov, K. Aderghal, J. Benois-Pineau, A. Krylov, G. Catheline, 3d cnn-based classification using smri and md-dti images for alzheimer disease studies (2018).

[35] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, E. Duchesnay, Scikit-learn: Machine learning in Python, Journal of Machine Learning Research 12 (2011) 2825–2830.