ON SELECTING IMAGES FROM AN UNAIMED VIDEO STREAM FOR PHOTOGRAMMETRIC MODELLING

P. Rönnholm [1, *], M.T. Vaaja [1], H. Kauhanen [1], T. Klockars [2]

[1] Department of Built Environment, Aalto University, P.O. BOX 14100, 00076 AALTO, Finland
– (petri.ronnholm, matti.t.vaaja, heikki.kauhanen)@aalto.fi

[2] Helsinki University Hospital, P.O. BOX 100, FI-00029 HUS, Finland – (tuomas.klockars)@hus.fi

Commission II, WG 4

KEY WORDS: unaimed video, object detection, convolutional neural network, voxel, structure-from-motion, imaging geometry

ABSTRACT:

In this paper, we illustrate how convolutional neural networks and voxel-based processing together with voxel visualizations can be utilized for the selection of unaimed images for a photogrammetric image block. Our research included the detection of an ear from images with a convolutional neural network, computation of image orientations with a structure-from-motion algorithm, visualization of camera locations in a voxel representation to detect the goodness of the imaging geometry, rejection of unnecessary images with an XYZ buffer, the creation of 3D models in two different example cases, and the comparison of resulting 3D models. Two test data sets were taken of an ear with the video recorder of a mobile phone. In the first test case, a special emphasis was taken to ensure good imaging geometry. On the contrary, in the second test case the trajectory was limited to approximately horizontal movement, leading to poor imaging geometry. A convolutional neural network together with an XYZ buffer managed to select a useful set of images for the photogrammetric 3D measuring phase. The voxel representation well illustrated the imaging geometry and has potential for early detection where data is suitable for photogrammetric modelling. The comparison of 3D models revealed that the model from poor imaging geometry was noisy and flattened. The results emphasize the importance of good imaging geometry.

1. INTRODUCTION

Photogrammetric 3D modelling typically requires many images taken from different perspectives. The current level of automation in commercial and freeware software has made photogrammetric techniques available to everybody. The key elements for such a level of automation have been automatic corresponding point search (Karami et al., 2017), structure-from-motion algorithm (Mouragnon et al., 2009; Westoby et al., 2012), and dense image matching (Remondino, 2014). However, the high level of automation might give the illusion that a successful model can always be made from any set of images. In some cases, it might be difficult to detect deformations of the resulting 3D model—especially if no reference data is available.

Solving exterior orientations of images and measuring of 3D points cannot be accurate without successful camera calibration. Automatic workflows provide an option that camera calibration is solved with self-calibration simultaneously with object point measurements. For accurate self-calibration, good imaging geometry is required, and in many cases imaging geometry for the reconstruction of an object is not optimal for comprehensive camera calibration (Remondino and Fraser, 2006). In addition, non-professionals are not necessarily aware of good camera network design (e.g., Fraser, 1984; Fraser, 1996), which easily leads to insufficient imaging geometry for camera calibration. The situation is even more difficult if a camera network is established randomly. This can be the case if images are taken automatically without knowing what objects will be of interest or if there is no possibility to properly aim the camera. In such cases, good imaging geometry can be especially difficult to ensure.

Automatic recognition of images that include interesting objects become useful if we cannot be sure that an object appears within the image footprint. Convolutional neural networks have become popular for object recognition tasks (Zhiqiang and Jun, 2017) even if there has been some doubts that they are the most efficient version of neural networks for this purpose (Hinton et al., 2011). Recently, convolutional neural networks have also become popular in the field of photogrammetry. The trend becomes visible when Google Scholar articles are searched for the keywords ‘photogrammetry’ and ‘convolutional neural networks’ (Fig. 1). The first article was published in 2008.

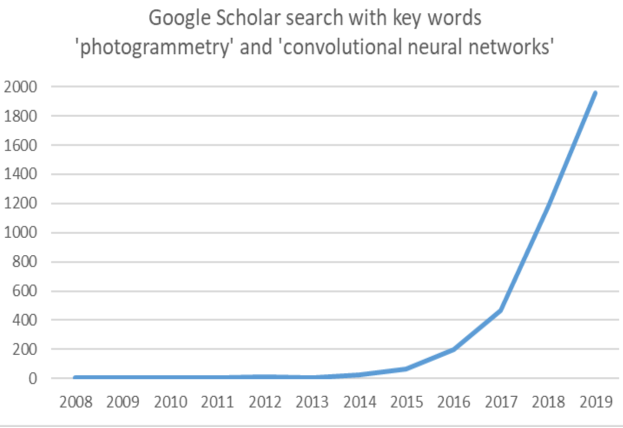


Figure 1. The annual number of photogrammetric publications utilizing convolutional neural networks reveals the popularity of the topic.

[*] Corresponding author

This contribution has been peer-reviewed. The double-blind peer-review was conducted on the basis of the full paper.
https://doi.org/10.5194/isprs-annals-V-2-2020-389-2020 | © Authors 2020. CC BY 4.0 License.

Even though convolutional neural networks were invented in the 1980s (Fukushima,1980; Waibel et al., 1989; LeCun et al., 1989), the method became practical in the 2000s when GPU processing (Chellapilla et al., 2006) and the efficient training of deep neural networks (Hinton et al., 2006) became available. Detailed presentations of deep convolutional neural networks for object recognition and image classification can be found in Liu et al. (2019) and Rawat and Wang (2017).

Voxels have gotten attention when processing 3D data. Numerous applications have utilized voxel structures. Recent examples of such applications are segmentation (Xu et al., 2017), point cloud registration (Wang et al., 2016), noise filtering (Rönnholm et al., 2015), and leaf area estimation (Itakura and Hosoi, 2019), just to name a few. The use of voxels can efficiently reduce data. In addition, the structure is regular, which enables efficient processing. In this paper, voxels are applied for both visualization and data reduction purposes.

Voxels have gotten attention when processing 3D data. Numerous applications have utilized voxel structures. Recent examples of such applications are segmentation (Xu et al., 2017), point cloud registration (Wang et al., 2016), noise filtering (Rönnholm et al., 2015), and leaf area estimation (Itakura and Hosoi, 2019), just to name a few. The use of voxels can efficiently reduce data. In addition, the structure is regular, which enables efficient processing. In this paper, voxels are applied for both visualization and data reduction purposes.

The aim of this paper is to illustrate how convolutional neural networks and voxel-based processing together with visualizations can be utilized for the selection of unaimed images for a photogrammetric image block. The voxel method has the potential to reveal whether imaging geometry is suitable for photogrammetric 3D modelling and can reduce a redundancy of images. In our test case, the operator's ear is the object in focus. However, the method applies to any object if data is collected in an unaimed manner.

2. MATERIALS AND METHODS

2.1 Data collection

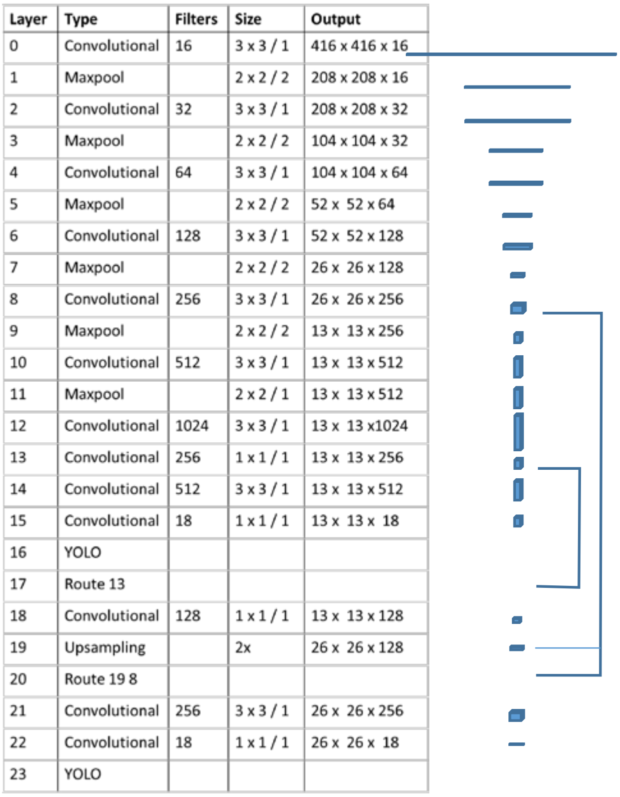
In this article, we arranged test cases where videos of an operator's ear are taken with a mobile phone. This task is difficult, because an operator cannot see a live-view during video acquisition making visual aiming impossible. Therefore, the aiming of the camera was based on feeling. We preferred to use the rear-facing camera of the phone because its image quality is better than the front-facing ('selfie') camera.

Two videos of the ear were taken with a Samsung s10e mobile phone's rear-facing camera. In addition, one additional video was taken for training a convolutional neural network. The Samsung s10e mobile phone actually has two rear-facing cameras. The normal lens, with a pixel size of 1.4 µm, was selected instead of the ultra-wide lens. Full HD videos (1920 x 1080 pixels) were taken at 30 frames per second. In the first case, video was taken considering good image geometry, which included images from all over the ear. In the second case, the camera was moved along a roughly horizontal trajectory simulating non-optimal imaging geometry. This case is not unrealistic, because if an elbow is not raised and lowered during image acquisition, the possible trajectory of a hand leads to such imaging geometry.

In both cases, the video recording was started by keeping the mobile phone in front of the operator to make it easier to hit the start button. Then the camera was moved to take the desired images. Finally, the camera was returned to the front of the operator to make it easy to stop the recording.

2.2 Training the convolutional neural network

We applied a YOLO (You Only Look Once) convolutional neural network (Redmon et al., 2016) for detecting whether the operator's ear was visible in the examined video frames. To train the weights of a convolutional neural network, we labelled 100 images of the operator's ear and utilized the yolov3-tiny model (Fig. 2). Images represented the ear from different viewing angles. This model is a smaller version of the full 106-layer (75 convolutional layers and 31 maxpool, route, up-sampling, and YOLO layers) yolov3 model, including only 23 layers (13 convolutional layers and 10 maxpool, route, up-sampling, and YOLO layers). We decided to use only 100 images for training because only one class and one type of an ear were included. Normally, the training phase requires thousands of images.



Layer	Type	Filters	Size	Output
0	Convolutional	16	3 x 3 / 1	416 x 416 x 16
1	Maxpool		2 x 2 / 2	208 x 208 x 16
2	Convolutional	32	3 x 3 / 1	208 x 208 x 32
3	Maxpool		2 x 2 / 2	104 x 104 x 32
4	Convolutional	64	3 x 3 / 1	104 x 104 x 64
5	Maxpool		2 x 2 / 2	52 x 52 x 64
6	Convolutional	128	3 x 3 / 1	52 x 52 x 128
7	Maxpool		2 x 2 / 2	26 x 26 x 128
8	Convolutional	256	3 x 3 / 1	26 x 26 x 256
9	Maxpool		2 x 2 / 2	13 x 13 x 256
10	Convolutional	512	3 x 3 / 1	13 x 13 x 512
11	Maxpool		2 x 2 / 1	13 x 13 x 512
12	Convolutional	1024	3 x 3 / 1	13 x 13 x1024
13	Convolutional	256	1 x 1 / 1	13 x 13 x 256
14	Convolutional	512	3 x 3 / 1	13 x 13 x 512
15	Convolutional	18	1 x 1 / 1	13 x 13 x 18
16	YOLO			
17	Route 13			
18	Convolutional	128	1 x 1 / 1	13 x 13 x 128
19	Upsampling		2x	26 x 26 x 128
20	Route 19 8			
21	Convolutional	256	3 x 3 / 1	26 x 26 x 256
22	Convolutional	18	1 x 1 / 1	26 x 26 x 18
23	YOLO			

Figure 2. The layers of the yolov3-tiny convolutional neural network. The size of an incoming image is 416 x 416 x 3.

For training, the 64-bit c++ version of the open source neural network Darknet (https://pjreddie.com/darknet/) was utilized with GPU support in the Windows 10 operation system. The pre-trained weights of yolov3-tiny.conv.11 were applied as a starting point. The average loss of training with respect to iteration rounds is illustrated in Fig. 3. After 8,300 iterations, the average loss was considered to be stable enough to select those weights as the final weights. It is worth mentioning that the resulting weights are suitable only for finding the operator's ear or similar because of the limited training set.

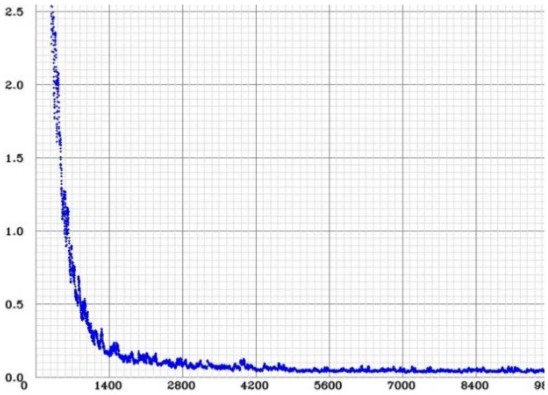


Figure 3. The average loss with respect to iteration rounds during the training phase.

2.3 Workflow

To reduce the number of images, we decided not to extract all frames. Instead, we set the number of target frames and computed a corresponding frame step. The step was estimated with

$$frame\ step = floor\left(\frac{total\ number\ of\ frames}{number\ of\ target\ frames}\right). \quad (1)$$

In the next step, we applied the python OpenCV implementation of YOLO with our trained weights for finding all frames that included an ear. Those frames that did not have an ear were rejected. As a result, we had only such images that were suitable for a photogrammetric 3D reconstruction of the ear.

The orientations of selected images were computed with the structure-from-motion algorithm. In our case, we utilized the freeware VisualSFM. The solution provides XYZ coordinates of the projection centres of images (i.e., camera locations), which were utilized in the next step. The rotations were not considered useful, because the previous step ensured that an ear was visible in all images.

Next, a coarse voxel grid was established on the top of the object in order to get an impression of the general imaging geometry. The orientation of the voxel grid was set manually. At the initial stage, each voxel included the number zero. The value of a voxel was set to 1 (or increased by one each time) if the projection centre of an image was found within a voxel. This examination reveals whether the imaging geometry has any major flaws, because the distribution is visualized with just a few, easily interpretable voxels. The size of voxels needs to be adjusted, depending on the case. In our case, we set voxels in such a way that the target was under the middle voxel and the total size of the grid was 5x5x5. Such a voxel system has the potential to indicate whether or not the imaging geometry is sufficient.

In order to remove images that were unnecessarily close to other images, another voxel grid with a smaller voxel size was created. In this paper, the voxel grid is called an XYZ buffer, because only one camera location is allowed to be within one voxel. The voxel size was adjusted so that the grid size becomes 11x11x11 voxels. Because the scale was not known at this point, the absolute voxel size was not defined. However, it can be estimated to be approximately 5–6 cm. Camera locations were compared with the centres of voxels. When several images appeared to be within the same voxel, we selected the one that was closest to the centre point of that voxel.

The remaining images in both test cases were processed in Agisoft Metashape in order to create a 3D model. These models were transferred in Geomagic Studio 11 for a 3D surface comparison. Because of the unknown absolute orientation, models were automatically aligned with the ICP surface matching tool of Geomagic Studio. After that, models were manually scaled to the correct scale by measuring the height of the ear with a ruler in order to get interpretable units of difference values.

3. RESULTS

In the first test case, YOLO rejected 23 images out of 103. It found all images in which an ear was completely visible. The orientation of an ear did not influence detection. If an ear was only partially visible, YOLO detected it correctly three times and did not recognize it four times. Fig. 4 shows two examples of a successfully detected ear with the corresponding boundary boxes.

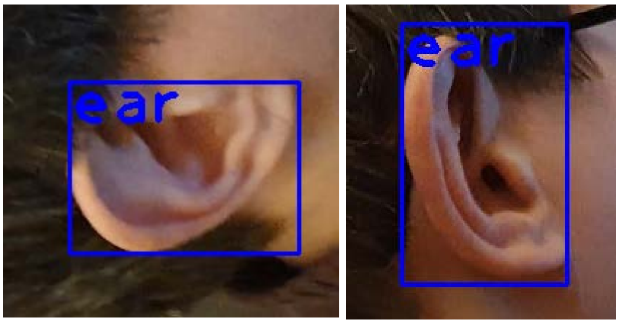


Figure 4. Two examples of how YOLO successfully found an ear from images.

The second video was shorter than the first one. Therefore, the total number of images was 68. In this case, YOLO found 50 ear images and rejected 18 images. If the ear was only partially visible, YOLO detected it two times and did not recognize it three times. In this case, there were two mistakes when YOLO rejected an image even if there actually was an ear visible. In all failed cases, the imaging perspective was from the back of the ear. See Fig. 5.

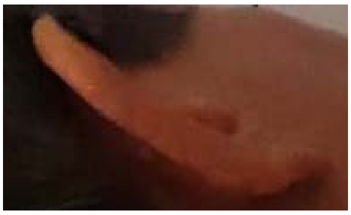


Figure 5. An example of how YOLO failed to find an ear from an image.

The images that YOLO accepted were processed in VisualSFM. The orientation results of the first experiment from the structure-from-motion algorithm are illustrated in Fig. 6. In this case, imaging geometry is relatively good, i.e. there are images from varying perspectives around the ear. The orientation results of the second case (Fig. 7) reveals how the trajectory is almost only horizontal.

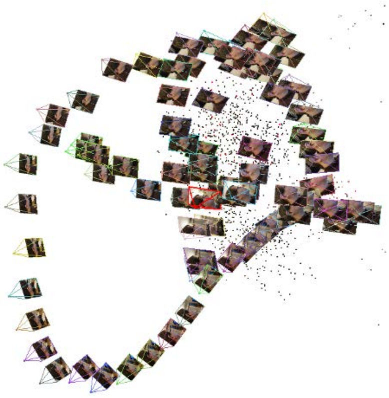


Figure 6. Imaging geometry of the first test case. Images were taken from varying perspectives, establishing good imaging geometry.

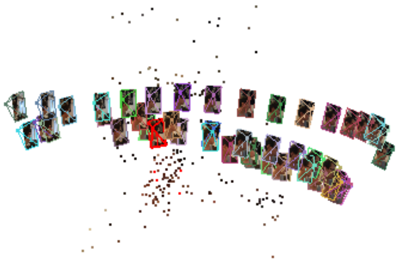


Figure 7. Imaging geometry of the second test case. Imaging geometry is not as strong as in the first test case because images were taken along a nearly horizontal trajectory.

Fig. 8 and Fig. 9 visualize 5x5x5 grids of the imaging geometry from the first and the second test cases, respectively. In each green voxel, there exists one or more camera locations. Oblique views reveal the 3D structure of the imaging geometry. The top views demonstrate how even such limited illustrations show a clear difference in imaging geometries.

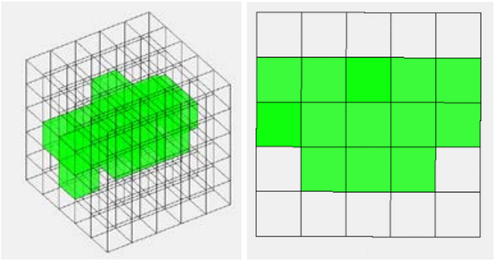


Figure 8. The oblique voxel representation (left image) and the top view (right image) of the imaging geometry in the first test case.

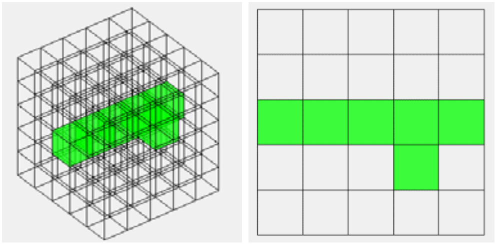


Figure 9. The oblique voxel representation (left image) and the top view (right image) of the imaging geometry in the second test case.

The number of images was reduced with a 11x11x11 –voxel XYZ buffer. Fig. 10 illustrates the oblique and top representations of the voxel grid in the first test case. Within each green voxel, there is only one accepted image location. The object lies approximately at the centre below the XYZ buffer. A similar illustration of the second case can be seen in Fig. 11. In the first test case, the XYZ buffer rejected 16 images out of 80. Correspondingly, in the second case, 11 images out of 50 were rejected.

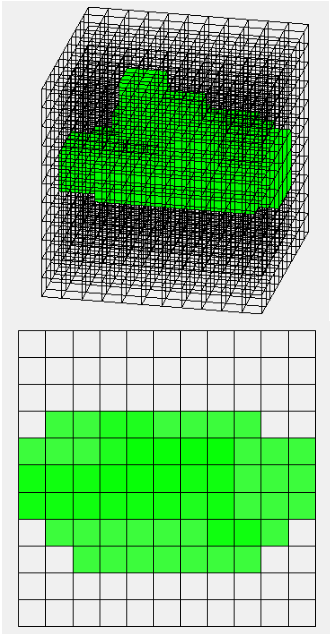


Figure 10. The oblique voxel representation (upper image) and the top view (lower image) of the XYZ buffer in the first test case.

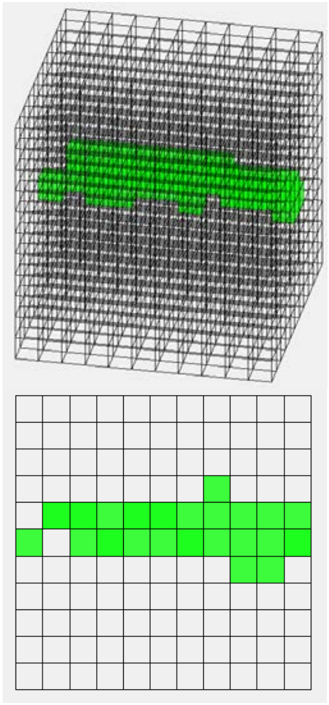


Figure 11. The oblique voxel representation (upper image) and the top view (lower image) of the XYZ buffer in the second test case.

In Fig. 12, the difference in the two test cases is illustrated by the 3D surface comparison. In this case, the model from the first test case is the reference to which another model was compared. Units are metres and the green colour indicates agreement between models. The areas illustrated with blue colours are closer to the viewer than they should be, and red areas are too far from the viewer. Notable peaks in differences are caused by several noise peaks of the model in the second test case. The range of differences was between -2.74 and +2.69 mm.

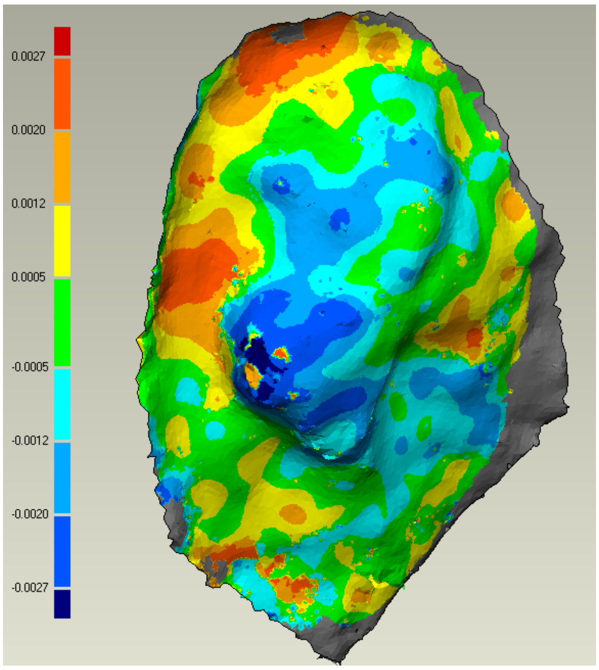


Figure 12. The surface comparison of models from the first and second test cases. Because of a nonoptimal imaging geometry in the second test case, the depth scale is incorrect, making the object flat. In addition, this model is noisy. The unit is metres.

4. DISCUSSION

If the main rear-facing camera is applied, the user cannot aim the camera by visual observations without a remote display. Difficult aiming easily causes the ear to be not visible in all frames. In the worst cases, this can negatively affect the imaging geometry. If an operator is not moving a camera, many frames are taken from approximately the same location. Therefore, the number of frames extracted from a video does not reveal how well the frames are distributed in the space. For example, the 39 accepted frames of the second test case would have been more than enough to model the ear if the distribution were optimal.

The YOLO convolutional neural network seems to find the ear well regardless of orientation, even if the amount of training data was very low. However, there were two failed cases. Afterwards the training data set was examined, and it appeared that there were too few reference images taken from the back of an ear, as in the failed cases. This emphasizes that a training data set should be as comprehensive as possible. To generalize the YOLO search to cover all kinds of ears would require thousands of labelled images representing ears of various shapes and orientations. In addition, the yolowv3-tiny model is most probably not deep enough for such variations in shape. In our example, YOLO detected some of the images where an ear was partially outside of the image area (Fig. 13), but not all of them. It is expected that a deeper convolutional neural network would also be better at finding partially visible objects. However, in our case there is not much value in using such images. Therefore, it is not critical to detect partially visible objects.

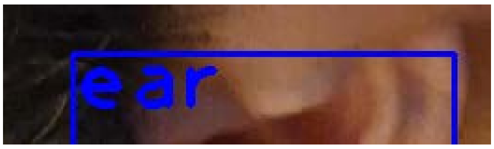


Figure 13. In some cases, YOLO detected an ear that was only partially visible within the image footprint.

A voxel representation well illustrates the imaging geometry. A coarse voxel grid should be enough to evaluate whether there are enough images from all perspectives. Finding camera orientations with the structure-from-motion is much faster than generating 3D models. Therefore, it can be valuable information to detect at an early stage if a 3D model can be expected to succeed. In this paper, the voxel grids were utilized only for visual illustrations. However, in fixed problems, one might utilize automation to detect whether there are observations in all required voxels.

Utilizing a voxel grid as an XYZ buffer for reducing the number of images seems to work well. In this case, the orientation of the grid is actually not important—even if we applied an intuitively easy orientation for visualization purposes. The selection of a grid size is case-dependent. In our example, we targeted a certain number of voxels. However, in some cases, absolute values could be more advantageous. In such cases, the scale of a system needs to be solved earlier than we did. The computation time of photogrammetric 3D modelling is dependent on the number of images. Therefore, unnecessary images should be removed. Images that are very close to each other do not add significant information to the imaging geometry of an image block.

The 3D surface comparison (Fig. 12) reveals that there are significant differences between models. Because processing in both cases was similar, the only difference is in imaging geometry. The first test case produced a 3D model that was very similar to the original object. However, in the second case with poor imaging geometry, the result was much worse. The results indicate that the model is at worst about 5.5 mm too flat. That is a significant error in the case of an ear. The reason for this phenomenon can be traced to the camera calibration and poor imaging geometry. A self-calibration with poor imaging geometry does not give acceptable interior orientation results. Therefore, it is strongly recommended to either pre-calibrate a camera and/or ensure that the imaging geometry is sufficient for camera calibration. In addition to flatness, the second model was very noisy compared to the first case.

5. CONCLUSIONS

We aimed to select a suitable set of images from an unaimed video for photogrammetric 3D modelling. Each image should include the desired object—in our case, an ear. In addition, we aimed to visualize the imaging geometry and to remove images that were close to other images.

Experiences with the YOLO convolutional neural network was positive. Almost all ear images were detected even if our training data was very small. However, more comprehensive training data is required to generalize ear detection to all types of ears.

Visualizations of camera locations in a 3D space with a coarse voxel representation have the potential to reveal the quality of imaging geometry at an early stage. In our paper, only visual interpretation was given. However, in future research, this may be developed into an automatic quality indicator of the imaging geometry.

The use of an XYZ buffer enables easy rejection of images that are too close to other images. In this way, the number of images in a photogrammetric block can be adjusted in such a way that there are less unnecessary images.

The comparison of two test cases revealed significant differences between final 3D models. Poor imaging geometry led to flattened and noisy results, whereas good imaging geometry provided a satisfying 3D model. This emphasizes the importance of imaging geometry.

ACKNOWLEDGEMENTS

The authors would like to thank the Academy of Finland for financial support through the projects, ‘Centre of Excellence in Laser Scanning Research (No. 272195)’ and the Strategic Research Council project COMBAT – ‘Competence-Based Growth Through Integrated Disruptive Technologies of 3D Digitalization, Robotics, Geospatial Information and Image Processing/Computing – Point Cloud Ecosystem (No. 293389)’.

REFERENCES

Bianco, S., Ciocca G., Marelli, D., 2018. Evaluating the Performance of Structure from Motion Pipelines. *Journal of Imaging,* 4(98), 18 pages.

Chellapilla, K., Puri, S., Simard, P., 2006. High performance convolutional neural networks for document processing. In *Proceedings of the 10th International Workshop on Frontiers in Handwriting Recognition*. Los Alamitos, CA: IEEE Computer Society, 6 pages.

Fraser, C.S., 1984. Network design considerations for non-topographic photogrammetry. *Photogrammetric Engineering and Remote Sensing*, Vol. 50(8), 1115-1126.

Fraser, C.S., 1996. Network Design. In K.B. Atkinson, editor, *Close Range Photogrammetry and Machine Vision*, Whittles Publishing, Roseleigh House, Latheronwheel, Caithness, KW5 6DW, Scotland, UK, 256-281.

Fukushima, K., 1980. Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological Cybernetics*, 36, 193-202.

Gorte, B. G. H., Zhou, K., van der Sande, C. J., Valk, C., 2018. A computationally cheap trick to determine shadow in a voxel model. *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, IV-4, 67-71.

Hinton G.E., Krizhevsky A., Wang S.D., 2011. Transforming Auto-Encoders. In: Honkela T., Duch W., Girolami M., Kaski S. (eds) Artificial Neural Networks and Machine Learning – ICANN 2011. ICANN 2011. *Lecture Notes in Computer Science*, vol. 6791. Springer, Berlin, Heidelberg, 44-51.

Hinton, G.E., Osindero, S., Teh, Y., 2006. A fast learning algorithm for deep belief nets. *Neural Computation*, 18(7), 1527-1554.

Itakura, K., Hosoi, F., 2019. Voxel-based leaf area estimation from three-dimensional plant images. *Journal of Agricultural Meteorology*, 75, 211-216.

Karami, E., Prasad, S., Shehata, M., 2017. Image Matching Using SIFT, SURF, BRIEF and ORB: Performance Comparison for Distorted Images. *arXiv preprint* arXiv:1710.02726, 5 pages.

LeCun, Y., Boser, B., Denker, J.S., Henderson, D., Howard, R.E., Hubbard, W., Jackel, L.D., 1989. Backpropagation applied to handwritten zip code recognition. *Neural computation*, 1(4), 541-551.

Liu, L., Ouyang, W., Wang, X., Fieguth, P., Chen, J., Liu, X., Pietikäinen, M., 2019. Deep Learning for Generic Object Detection: A Survey. *International Journal of Computer Vision*, https://doi.org/10.1007/s11263-019-01247-4, 58 pages.

Mouragnon, E., Lhuillier, M., Dhome, M., Dekeyser, F., Sayd, P., 2009. Generic and real-time structure from motion using local bundle adjustment. *Image and Vision Computing*, 27(8), 1178-1193.

Rawat, W., Wang, Z., 2017. Deep Convolutional Neural Networks for Image Classification: A Comprehensive Review. Neural Computation, 29, 2352-2449.

Redmon, J., Divvala, S., Girshick, R., Farhadi, A., 2016. You only look once: unified, real-time object detection, In CVPR, 2016, arXiv preprint, arXiv: 1506.02640, 10 pages.

Remondino, F., Spera, M.G., Nocerino, E., Menna, F., Nex, F., 2014. State of the art in high density image matching. *Photogrammetric Record*, 29, 144-166.

Rönnholm, P., Kukko, A., Liang, X., Hyyppä, J., 2015. Filtering the outliers from backpack mobile laser scanning data, *The Photogrammetric Journal of Finland,* 24(2), 20-34.

Waibel, A., Hanazawa, T., Hinton, G., Shikano, K., Lang, K.J., 1989. Phoneme recognition using time-delay neural networks. In *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 37(3), 328-339.

Wang, J., Lindenbergh, R, Shen, Y., Menenti, M., 2016. Coarse point cloud registration by EGI matching of voxel clusters. *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences,* XXIII/III-5, 98-103.

Westoby, M.J., Brasington, J., Glasser, N. F., Hambrey, M. J., Reynoldsc, J.M., 2012. ‘Structure-from-Motion’ photogrammetry: A low-cost, effective tool for geoscience applications. *Geomorphology,* 179(15), 300-314.

Xu, Y., Hoegner, L., Tuttas, S., Stilla, U., 2017. Voxel- and graph-based point cloud segmentation of 3D scenes using perceptual grouping laws. *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, IV-1/W1, 43-50.

Zhiqiang, W., Jun, L., 2017. A review of object detection based on convolutional neural network. In *Proceedings of the 36th Chinese Control Conference*, Dalian, China, 11104-11109.