

## SUPPORTING THE MANAGEMENT OF HUMANITARIAN OPERATIONS CONCERNING MIGRATION MOVEMENTS WITH REMOTE SENSING

L. Wickert<sup>1,\*</sup>, M. Bogen<sup>1</sup>, M. Richter<sup>1</sup>

<sup>1</sup> Fraunhofer IAIS, Schloss Birlinghoven, 53757 Sankt Augustin, Germany  
- (lorenz.wickert, manfred.bogen, marvin.richter)@iais.fraunhofer.de

**KEY WORDS:** Monitoring and Management, Remote Sensing (RS), Artificial Intelligence (AI), Machine Learning (MS), Convolutional Neural Networks (CNNs), Faster R-CNN, Object Detection, Dwelling Detection

### ABSTRACT:

The various forms of humanitarian operations include operations concerning the management of migrant movements and refugees. Managing those operations is non-trivial. A large number of refugees have to be welcomed, registered, forwarded, and be given supplies and accommodation. This is due to a lack of current and sufficient information about the refugees, making planning and execution of operations challenging, expensive and cumbersome. The earlier information about the refugees is available, the better. The method “Dwelling Detection”, conducted on satellite imagery of refugee camps, can provide large-scale heads-up information fast, complementing information already available to operators at the ground. With “Dwelling Detection”, dwellings in a camp and their extent are detected using machine learning methods. An estimate of inhabitants of the camp is computed using the number and the extent of the detected dwellings. Our workflow uses a Faster R-CNN, an object detection network. To train the network, we developed a fast training data annotation workflow. We use the dwellings detected by the faster R-CNN to estimate a number of inhabitants. The quality of the analysis can be evaluated by a confidence-metric, computed out of the results of the Faster R-CNN. The results can be used in humanitarian operations. We tested the workflow using different configurations and data. From those tests, we give recommendations on how to build a dwelling detection classifier. We propose to humanitarian operators to build a dwelling detection classifier according to our recommendations and use satellite images in actual humanitarian operations. This could help to reduce stress for all people involved in a humanitarian (crisis) situation.

### 1. INTRODUCTION

In 2015/16, the biggest migration movements to Germany after the Second World War occurred. The civil war in Syria and the terror organization Islamic State (IS) mainly caused these migration movements. While being high throughout the whole year, the number of refugees arriving in Germany and Austria intensified in September 2015 after the Dublin Regulation was deferred. At peaks, more than ten thousand refugees were crossing the Austrian-German border a day. Since then, the global situation has not relaxed. The United Nations High Commissioner for Refugees reported that in 2018 “the worlds forcibly displaced population remained yet again at a record high” (“Global Trends - Forced Displacement in 2018” 2018). In 2020, the refugee situation at the Greek-Turkey border is tensing (Rankin et al., 2020).

Due to a short reaction time and the high intensity of the migration movements, the humanitarian and administrative challenges are high during humanitarian operations. Refugees have to be welcomed, registered, forwarded and be given supplies and accommodation. Problems are caused by the high number of people as well as by the lack of quickly available information, making humanitarian and administrative operations challenging, expensive and cumbersome. As a complement to the information already available at the borders themselves, satellites can provide large-scale heads-up information fast. When dealing with a migration situation, satellite images can give information about where refugees are way before they arrive at a border giving first responders urgently needed lead-time.

Dwelling Detection (Spröhnle et al., 2014), conducted on satellite images of refugee camps, count dwellings in a camp even far away from a border. Due to advances in satellite sensors (Campbell, Wynne, 2011, p.187ff) and object detection using machine learning (Zhao et al., 2019), this task can be automated. In Dwelling Detection, the number of inhabitants in a camp can be estimated from the found dwellings, given a knowledge of how many persons fit into dwellings of different sizes. To count the dwellings, object detection machine learning methods can be used.

In this paper, a dwelling detection workflow using a Faster R-CNN (Ren et al., 2017), used for object detection, is described. To train the Faster R-CNN, we developed a fast training data annotation workflow. The Faster R-CNN finds bounding boxes around dwellings and outputs an estimate of people living in a camp, derived from the number and the extend of the found dwellings. The workflow produces results useful for humanitarian operations of the future, when immediate estimates are required, helping the authorities and organizations managing a humanitarian operation on one side, and the refugees themselves on the other side. After developing a workflow, we conducted various tests: The training data annotation workflow was tested using two different workflow configurations (see Figure 1), different results for colour- vs. greyscale- image-analysis were compared and generalisation was tested using a second set of satellite images (see Figure 2). From this, we derived recommendations concerning the training of a deep neural network dwelling detection classifier to be used in a dwelling detection workflow.

The research described in this paper was conducted in the con-

\* Corresponding author

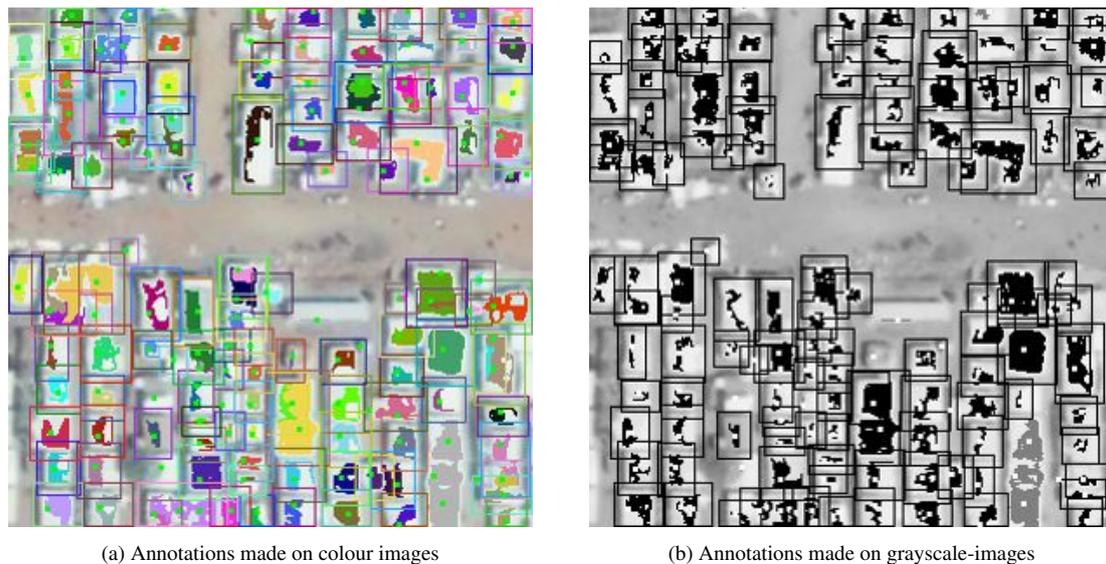


Figure 1. Example of the two annotation modes. The annotations shown are made on a satellite image of the Juba camp, South Sudan

text of the HUMAN+<sup>1</sup> project. In the HUMAN+ project, a real-time situational awareness system for efficient management of migration movements was developed. Besides the dwelling detection module, camera streams from cameras located at borders are analyzed, social media platforms are evaluated concerning migration movement and reports from operators in the field are collected and included in the situational awareness system.

## 2. METHODOLOGY

In the following, we describe the approach “Dwelling Detection” in Section 2.1, the training process in Section 2.2 and the final Dwelling Detection workflow in Section 2.4. Those sections are cited from (Wickert et al., 2020). Section 2.3 describes extensive tests of the achieved Dwelling Detection classifier, evaluating the performance and behaviour of various configurations.

### 2.1 Deep Learning Dwelling Detection

Because individual humans cannot be seen on commercially available satellite images, Dwelling Detection, conducted on very high-resolution (VHR) satellite images of refugee camps offers a fitting method for extracting valuable information about humans from satellite imagery. Dwellings are counted and multiplied with an average, size-based factor of how many people live in one dwelling. From this, an estimate over how many people may live in a camp can be made. This information can be used in medium-term planning of humanitarian operations during migration situations, corresponding to the assessment phase in the UNHCR operations management cycle. In the assessment phase, “needs and the scale of the response required” are identified (UNHCR, 2015). Reasons why these information are not available for humanitarian operators include that camps are run by private companies (Katz, 2016), that camps do not develop as planned because of a rapid growth of inhabitants (Dalal et al., 2018) or because camps are makeshift camps which are created arbitrarily, sometimes called “jungles”; (Katz, 2016) and (Beznec et al., 2016).

<sup>1</sup> <https://giscience.zgis.at/human/> (20 April 2020)

Experts traditionally do dwelling Detection by hand, taking a lot of time which is not available in a crisis situation. Recent advantages in image classification and object detection on images using Convolutional Neural Networks (CNN) allow first approaches to automate this monotonous task; (Quinn et al., 2018) and (Ghorbanzadeh et al., 2018). A Faster R-CNN offers a state-of-the-art CNN architecture for object detection to eventually develop a Dwelling Detection Workflow. To do so, a complete machine learning workflow was created, following a framework of preparing input data, defining the expected output data and building a core network which constructs the intrinsic and natural relationship of the input-output pair (Zhang et al., 2016).

The workflow consists of three main steps. In the first step, we use satellite images, which are annotated by hand to prepare ground truth training data consisting of a train- and a validation-set. We train a Faster R-CNN with those sets. In the last step, the trained Faster R-CNN analyzes a new satellite image of a camp. The classifier outputs a dwelling count and a people estimate, a confidence value concerning the Faster R-CNN analysis, and the processed input satellite image with found dwellings marked.

### 2.2 Faster R-CNN Training

The input data in the described Dwelling Detection workflow is VHR satellite imagery of refugee camps, identified at present by a human being, taken from Google Earth (Google Inc.) due to the lack of freely available satellite data. On those, the dwellings forming a camp have to be found by an object detection algorithm like Faster R-CNN and marked with a bounding box.

A Faster R-CNN is trained using supervised learning. The ground truth data required to train the network consists of bounding boxes around each dwelling. Further, the size of the input images needs to be small enough to be efficiently handled by a Faster R-CNN. Therefore, the input satellite image is split up into 300 pixel x 300 pixel tiles. For speeding up the annotation process, we developed a workflow building up on a seeded region growing algorithm (Adams, Bischof, 1994). Each dwelling needs to be point-annotated by a human; then each annotation is used by the region growing algorithm as a seed to estim-



(a) Image trained on



(b) New image used for validation

Figure 2. Example of one image (a) trained on and used in validation sets `v_orig` and the new image (b) used in the validation sets `v_new`. Both images show the refugee camp Juba in South Sudan

ate the extent of one dwelling. The results of the region growing algorithm are then filtered to eliminate bad regions. Around each region, a bounding box is defined and stored in the MS COCO-annotation format (Lin et al., 2014).

We used satellite images of nine different refugee camps to create training sets. The images have a huge variation in the landscape surrounding the camps, the organization form of the camps, the size of the camps and the quality of the images caused by the recording distance of the satellite and the satellites optical sensors. On each image, annotations on tiles of the upper 50 % of the input image are added to a training set, annotations on tiles on the lower 50 % of the input image are added to a validation set.

For implementation and training of the Faster R-CNN, we used

an open source implementation (Girshick et al., 2018). A Faster R-CNN consists of a pre-trained backbone network for feature extraction, a Regional Proposal Network (RPN) for generating object proposals and a classification network outputting bounding boxes around found objects as well as a class vector for each bounding box. As a backbone, we used ResNet50 (He et al., 2016). We trained the initial network on one GPU for 36,000 iterations (epochs).

### 2.3 Test and Validation Workflow

We tested our workflow on various configurations using different parameters to get a better knowledge about the strengths and limitations of the developed classifier. Therefore, we trained various Faster R-CNNs using different configurations. We evaluated the resulting Faster R-CNNs on several validation sets.

Four parameters were varied during the tests. The first parameter were the input satellite images that were analysed by the networks. The second parameter concerned the configuration of the training data annotation workflow, for both training and validation. The third parameter was the training time of the trained networks. Finally, the colour-representation of the images trained with and the images that were analysed was varied using greyscale- and colour-images.

Conf	Training On	Annotations On	Train Time
n_cc_36k	Colour	Colour	36,000 Iter.
n_gc_27k	Grey	Colour	27,000 Iter.
n_gc_36k	Grey	Colour	36,000 Iter.
n_gg_27k	Grey	Grey	27,000 Iter.
n_gg_36k	Grey	Grey	36,000 Iter.

Table 1. Training configurations of the networks tested

In total, we trained five Faster R-CNNs. The parameters varied were the colour-representation of the training set, the configurations of the annotation creation workflow with annotations made on colour- or greyscale-images, and the training time for each network (see Table 1). Network n\_cc\_36k, the initial network, was trained for 36,000 iterations on colour-images with annotations made on colour images. Two networks (n\_gc\_27k & n\_gc\_36k) were trained on greyscale-images with annotations made on colour-images with a training time of 27,000 or 36,000 iterations, respectively. Further, two networks (n\_gg\_27k & n\_gg\_36k) were trained on greyscale-images with annotations made on greyscale-images with a training time of 27,000 or 36,000 iterations, respectively.

To validate the results of the different networks, we created in total six validation sets ordered in two sets of each three validation sets (see Table 2). The first set (v\_orig) contains validation sets based on the images the networks were trained on. They were created during the regular training data annotation workflow. For the validation sets of the second set (v\_new), we annotated new satellite images that the networks have not seen before. These images show the same camps as the initial images but were shot at different times and environmental conditions. It has to be noted, that only some parts of the new satellite images have been annotated, resulting in fewer but exemplary tiles per image added to a validation set. In average around 200 dwellings were annotated per camp. The three validation sets in one set differ again in the colour-representation of the images contained and the colour-representation on which the annotations were made on (see Figure 3).

Conf	Type	Images	Annotations On
v_orig_cc	Trained On	Colour	Colour
v_orig_gc		Grey	Colour
v_orig_gg		Grey	Grey
v_new_cc	Not Seen	Colour	Colour
v_new_gc		Grey	Colour
v_new_gg		Grey	Grey

Table 2. Configurations for the Validation Sets the different networks were trained on

We tested each Faster R-CNN on each Validation Set. For each test, Average Precision (AP) and Average Recall (AR) were calculated. Average Precision “summarises the shape of the precision/recall curve” where “recall is defined as the proportion of all positive examples ranked above a given rank” and precision is “the proportion of all examples above that rank which are from the positive class” (Everingham et al., 2010). AP and AR were calculated using the MS COCO detection evaluation

metric, where “AP and AR are averaged over multiple Intersection over Union (IoU) values” and “AP is averaged over all categories”. Further “AR is the maximum recall given a fixed number of detections per image, averaged over categories and IoUs” (COCO Consortium, 2015). For AR, the COCO-metric  $AR^{max=100}$  was used. Further, the F1 score, which is the harmonic mean of Precision and Recall was calculated. F1 is defined as (Chinchor, 1992):

$$F_1 = \frac{(\beta^2 + 1.0) * P * R}{\beta^2 * P + R} \quad (1)$$

where  $P$  = Precision  
 $R$  = Recall  
 $\beta$  = Weight for Precision

We set  $\beta = 1$  for the following tests to weight Average and Precision equally.

## 2.4 Dwelling Detection Workflow

To analyse a satellite image using the trained Faster R-CNN, a copy of the input image, which is cut in various 300 pixel x 300 pixel tiles to handle its size, is made. Each tile is analysed separately. Found objects are accepted as a dwelling if their class security for being a dwelling is higher than a defined threshold. This threshold for a found object being accepted as a dwelling was set to a class security of 0.85, following (Ghorbanzadeh et al., 2018).

The Dwelling Detection workflow has two outputs: The first output is a copy of the input satellite image with all found dwellings marked with a bounding box. The second output consists of the number of found dwellings and the estimated number of inhabitants in a camp. The number of inhabitants is estimated with regard to the size of the found bounding boxes. To be flexible to changing spatial resolutions on images made by different satellites, the estimation algorithm has to be independent of this information. We achieve this by assigning a fixed number of inhabitants to the smallest and the largest dwelling found by the Faster R-CNN. For the dwellings with sizes between those anchor points, the number of inhabitants in a tent is interpolated linearly.

From our HUMAN+ project partner Johanniter Austria we got the information, that in organized camps around ten people live in a standard tent when the camp is a transit camp while in a permanent camp around six people live in a tent. To encompass these information and to further acknowledge differentiating numbers of inhabitants in tents in unorganized camps we set the number of inhabitants for the smallest dwelling found on an image to three and the number of inhabitants for the largest found dwelling on an image to twelve.

Further, a confidence metric indicating the certainty of the network about the results of the analysis is calculated. The metric is the ratio of the number of found objects  $O_{cls} \geq 0.85$  with a class security higher or equal than 0.85 divided by the number of objects  $O_{cls} \geq 0.5$  with a class security higher or equal than 0.5.

## 3. RESULTS

A complete dwelling detection machine learning workflow was implemented, tested and validated. In section 3.1, we present the results of the tests and validations described in 2.3. In section 3.2, we discuss the machine learning and dwelling detection results. This section is cited from (Wickert et al., 2020).

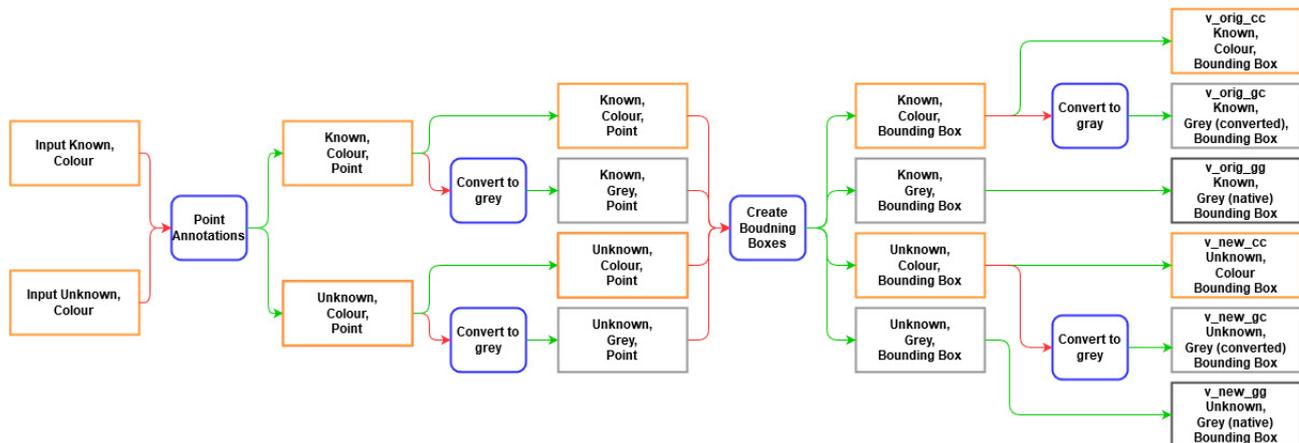


Figure 3. Workflow for creating the validation sets used for testing

### 3.1 Test and Validation Results

Table 3 shows the performance of the networks on images they were trained on. Table 4 shows the performance on images that the networks have not seen before. Looking at the results, various observations and from these, assumptions about the performance of a dwelling detection classifier can be made. From the assumptions, we formulate hypotheses about the training process and the performance of a Faster R-CNN when used for dwelling detection. From these hypotheses, we derived concrete Dos and Don'ts for building a deep learning Dwelling Detection classifier.

Looking at Table 3, the most obvious and expectable observation is that the best results for one validation set are obtained by the networks trained on training data created with the same configuration. Network n\_cc\_36k yields the best results for validation set v\_orig\_cc, network n\_gc\_27k and n\_gc\_36k yield the best results for validation set v\_orig\_gc and network n\_gg\_27k and n\_gg\_36k yield the best results for validation set v\_orig\_gg. This just shows the fundamental functionality of deep neural networks, that when training a deep neural network, it learns a representation of the input data (LeCun et al., 2015).

Besides the obvious, two observations can be made. The first observation is that networks trained on colour images achieve a higher maximum precision while networks trained on greyscale images achieve a better generalization over different validation sets. Looking at Table 3, the highest average precision achieved is 68,2% on validation set v\_orig\_cc by network n\_cc\_36k, which was trained on colour images. This is a 3% better AP-value than for network n\_gc\_36k, which was trained using the same annotations but on greyscale images, on its native validation set v\_orig\_gc. This is further exemplified when looking at the F1 score of network n\_cc\_36k on validation set v\_orig\_cc, which is also around 3% higher than that of network n\_gc\_36k on validation set v\_orig\_gc. We conclude, that colour enables learning more detailed features while training. On the other hand, looking at validation results for unknown images (v\_new) in Table 4, the networks trained on greyscale-images (networks n\_g) achieve better results than network n\_cc\_36k trained on colour images, which has the lowest F1 score on each validation set. This indicates that networks trained on greyscale images learn a more general representation of the input training data and therefore generalize better on unknown input data. A possible explanation for that behaviour could be found in (Geirhos et al., 2019): The authors show in experiments that

CNNs tend to learn textures of objects instead of object shapes. Making textures more abstract by using greyscale images could force the networks to focus learning on object shapes instead of object textures, boosting generalization.

The second significant observation is that training networks longer does not improve performance and can lead to overfitting (Srivastava et al., 2014) fast. An exemplary case is network n\_gg\_36k in Table 3. It outperforms its shorter trained version network n\_gg\_27k only on validation set v\_orig\_gg, the validation set build using the same configurations as the training data used to train networks n\_gg\_27k and n\_gg\_36k. On v\_orig\_gg, the F1-score of network n\_gg\_27k is significantly lower than that of network n\_gg\_36k. On validation set v\_orig\_gc, both networks have the same performance, both in AP and AR and in validation set v\_orig\_cc, the validation set with data the most different from the the networks training data, network n\_gg\_27k slightly outperforms network n\_gg\_36k. The better generalization properties of shorter trained networks can be further validated looking at the results in Table 4: Network n\_gc\_27k and n\_gg\_27k yield better results with continuously higher F1-scores on every validation set than their longer trained versions Networks n\_gc\_36k and n\_gg\_36k.

Concerning the quality and the differences between the annotation methods, we made only observations and we could not draw conclusions. Concerning annotations made on greyscale images, it can be noted that in Table 3, the networks trained using these annotations (n\_gg\_27k and n\_gg\_36k) achieve a significantly lower maximum AP and AR on their original validation set v\_orig\_gg than the networks trained using annotations made on colour images. Still, when looking at those networks unfamiliar validation sets v\_orig\_cc and v\_orig\_gc, they achieve better results as Networks n\_cc\_36k, n\_gc\_27k & n\_gc\_36k on their unfamiliar validation set v\_orig\_gg, with an AP around 10% higher and an AR around 15% higher. For generalization still, the colour representation of the training data seems to be of higher importance than the annotation method. In Table 4, network n\_cc\_36k achieves the worst results in all categories (except in AR at validation set v\_new\_cc) while the networks trained using greyscale training data have continuously similar results which are better than those of network n\_cc\_36k. Another interesting observation concerning annotations made on greyscale images is, that all networks achieve better results on validation set v\_new\_gg, containing annotations made on greyscale images than on validation set v\_new\_gc with annotations made on colour-images.

Validation Set	v_orig_cc			v_orig_gc			v_orig_gg		
	AP	AR	F1	AP	AR	F1	AP	AR	F1
Network									
n_cc_36k	68,2%	72,0%	70,0%	55,9%	61,1%	58,4%	27,6%	35,0%	30,9%
n_gc_27k	57,9%	63,4%	60,5%	60,9%	65,8%	63,3%	23,7%	30,4%	26,6%
n_gc_36k	61,6%	66,2%	63,8%	65,2%	69,1%	67,1%	23,6%	30,9%	26,8%
n_gg_27k	36,0%	48,2%	41,2%	35,4%	47,7%	40,6%	42,6%	48,1%	45,2%
n_gg_36k	35,5%	47,8%	40,7%	35,5%	47,9%	40,8%	46,2%	50,6%	48,3%

Table 3. Validation Results on the images trained on

Validation Set	v_new_cc			v_new_gc			v_new_gg		
	AP	AR	F1	AP	AR	F1	AP	AR	F1
Network									
n_cc_36k	6,6%	13,0%	8,8%	5,3%	11,7%	7,3%	9,2%	14,9%	11,4%
n_gc_27k	8,4%	12,8%	10,1%	8,8%	13,3%	10,6%	10,1%	14,4%	11,9%
n_gc_36k	8,2%	12,5%	9,9%	8,3%	12,8%	10,1%	9,8%	13,7%	11,4%
n_gg_27k	7,9%	13,9%	10,1%	8,4%	14,6%	10,7%	11,1%	17,2%	13,5%
n_gg_36k	7,3%	12,7%	9,3%	7,7%	13,4%	9,8%	9,7%	15,4%	11,9%

Table 4. Validation Results on unknown images

### 3.2 Dwelling Detection Results

We tested and trained the Dwelling Detection Workflow using the initial Faster R-CNN (network n\_cc\_36k) on nine satellite images of refugee camps with different sizes, from different parts of the earth, on different landforms and with varying image quality.

To estimate the number of inhabitants in a camp we used the workflow and parameters described in Section 2.4, which is independent of the spatial resolution of the input satellite image and therefore works without image-specific configurations.

To evaluate the estimate numbers, real-world numbers of inhabitants for each camp, measured at around the time the satellite image was shot, were researched in newspaper articles. It has to be noted that these numbers are not official numbers and are therefore fuzzy. Further, the estimations are based on no concrete knowledge of an individual camp. Therefore, the absolute numbers calculated by our Dwelling Detection workflow cannot be assumed as the real numbers, but function as an early warning system, stating an order of magnitude of the number of inhabitants.

Consequences from the result of the workflow have to be taken in accordance to the computed confidence metric and the visual output of the network. The confidence metric makes a statement about how well the Faster R-CNN could work with the input image. A low confidence means that there were many objects the network did not discard in the first place, but the network is also not sure about the objects being a dwelling. In that case, we recommend humanitarian operators to gather more information from different sources concerning the camp and the surrounding area. A higher confidence indicates that humanitarian operators can include the order of magnitude of displaced peoples in their medium-term planning.

## 4. DISCUSSION

In section 4.1, we discuss the operability of the general Dwelling Detection workflow citing (Wickert et al., 2020). From the networks behaviour in the tests and validation, we derive recommendations concerning the development of Dwelling Detection classifiers in section 4.2. In section 4.3, we give a critical assessment of the developed techniques concerning the usage in real humanitarian operations, following (Wickert et al., 2020).

### 4.1 Dwelling Detection Workflow

We developed the dwelling detection workflow with the goal of fast applicability. This goal was reached successfully by using open software and data, speeding up the slow and cumbersome process of ground-truth generation and annotation and building convenient workarounds when information about the data was missing. Nevertheless, there are tradeoffs between accuracy and applicability:

The seeded region growing algorithm used for speeding up annotating ground truth data can be improved using modern image segmentation algorithms (Zhu et al., 2016). A more accurate ground truth segmentation would also allow switching the deep learning architecture from a Faster R-CNN to a Mask R-CNN (He et al., 2017). Mask R-CNN builds up on the Faster R-CNN architecture and allows image segmentation on top of object detection in images. Further accuracy can be achieved by using georeferenced satellite imagery where the spatial resolution of a pixel is known. Combining a pixel-based image segmentation with the spatial resolution of a satellite image would allow calculating the actual size of a dwelling.

### 4.2 Faster R-CNN behaviour

From the tests and their results, we can extract some recommendations for training a deep learning dwelling detection classifier. Most importantly, we recommend having a high variance in training data with different types of camps and landscapes and satellite data with differing image quality, spatial resolution and environmental conditions. Especially in Dwelling Detection, the problem of camp types is emerging. Wild camps as well as organized camps need to be analysed with the same quality by a classifier, besides them having highly differing characteristics. A balance between planned and unplanned camps in the training set is essential. The more variance is contained in the training set, the more variant and adaptive the classifier can be.

When having a diverse training set, it is beneficial using colour- and greyscale-images. As shown in section 3.1, greyscale images are needed for achieving generalization in a classifier while the additional information offered by a colour-representation help building a more precise classifier.

Further, the problem of overfitting has to be tackled. The tests in Section 3.2 show that overfitting is a real problem that can

occur fast. To handle it, several measures can be taken. On the one hand, a high variance training set comes in handy. As discussed, it is important to have satellite imagery of different quality and showing different camps. It is important, even if it might feel counterintuitive at first, to not only have images of high but also of low (image) quality in the training set. When creating a training set, the focus should lie on acquiring many satellite images of different camps, i.e. on diversity, instead of annotating as much dwellings as possible on few satellite images. In this context, it is better to have fewer annotations from one camp but an overall high number of camps in the training set than the other way around. When annotated, the training set can be further enhanced using data augmentation techniques (Mikołajczyk, Grochowski, 2018). In the last step, while training a network, the training has to be monitored. Techniques like early stopping (Yao et al., 2007) should be used.

Concerning the annotation process, we can give no direct recommendations. The described annotation process is a trade-off between precision and fast operability. To clarify which of these two parameters is more important, the scope and purpose of a Dwelling Detection deep learning project has to be defined. When time is critical, the described method offers a great tool to develop a classifier fast. When time and manpower is not critical, it can be beneficial to create ground truth annotations by hand.

### 4.3 Critical Assessment

Whether accuracy or fast applicability is more important is context dependent. For operators managing one specific camp, results that are more accurate are essential. For humanitarian operators working on broader migration situations, fast information giving insights about the magnitude of the situation are important. In this context, the presented workflow is a good enhancement: We presented actual workflow results using the initial network n.cc\_36k in September 2019 to migration management agencies like the Red Cross as part of an international exercise carried out in the HUMAN+ project. The additional information about people in the refugee camps generated through Dwelling Detection was appreciated unanimously. Though those experts wanted to have absolute numbers, this is not doable due to the uncertainties described in this paper. The number of people in a refugee camp has to be seen in context with the confidence value generated. Those numbers have to be combined and merged with other information gathered in the HUMAN+ project such as the analysis of social media. Only then, a meaningful picture of the situation is achievable to help migration management agencies and first responders to make the right decisions in crisis management and refugees to be taken care of.

For developing new classifiers, the recommendations in Section 4.2 are in place. The development of the training set should be handled with care. The introduced fast annotation method leaves room for research, both concerning its behaviour during supervised learning and possible enhancements of the algorithm to increase its precision. When training new networks, the prevention of overfitting for a generalized classifier is crucial.

## 5. CONCLUSION

We built a machine learning based dwelling detection classifier using modern object detection techniques, a classifier which

yields results almost immediately helpful in humanitarian operations. An important step for doing so is our training data annotation workflow, which is fast and convenient. We tested and validated the resulting Dwelling Detection workflow, examining the behavior of the trained Faster R-CNNs and formulating recommendations for further deep learning Dwelling Detection projects.

The workflow outputs results in the magnitude of the real numbers of inhabitants in a migrant camp. It works on the image without the need of adjustments by the user, thus generating rapid and useful information. The generated information are best combined with other information about the migration situation from different sources. The visual representation of the results with bounding boxes drawn on the input image as well as the calculated confidence factor increase the interpretability of the results. The results were deemed to be useful by experts in the field. Further improvements in the various parts of the workflow are possible. The recommendations drawn from various tests can help building better classifiers and hereby provide better numbers for humanitarian operations in the future.

## ACKNOWLEDGEMENTS

The authors acknowledge the financial support by the German Federal Ministry of Education and Research (BMBF) and the Federal Ministry Republic of Austria Climate Action, Environment, Energy, Mobility, Innovation and Technology (BM-VIT/BMK) in the framework of the HUMAN+ project (funding code 13N14716 to 13N14723)

## REFERENCES

- Adams, R., Bischof, L., 1994. Seeded region growing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(6), 641–647. <http://ieeexplore.ieee.org/document/295913/>.
- Beznec, B., Speer, M., Mitrović, M. S., 2016. *Governing the Balkan route: Macedonia, Serbia and the European border regime*. Rosa Luxemburg Stiftung Southeast Europe.
- Campbell, J. B., Wynne, R. H., 2011. *Introduction to Remote Sensing, Fifth Edition*. Guilford Press.
- Chinchor, N., 1992. Muc-4 evaluation metrics. *Proceedings of the 4th Conference on Message Understanding, MUC4 '92*, Association for Computational Linguistics, USA, 22–29.
- COCO Consortium, 2015. COCO - Common Objects in Context - Detection Evaluation. <http://cocodataset.org/#detection-eval> (20 April 2020).
- Dalal, A., Darweesh, A., Misselwitz, P., Steigemann, A., 2018. Planning the Ideal Refugee Camp? A Critical Interrogation of Recent Planning Innovations in Jordan and Germany. *Urban Planning*, 3(4). <https://www.cogitatiopress.com/urbanplanning/article/view/1726>.
- Everingham, M., Van Gool, L., Williams, C. K. I., Winn, J., Zisserman, A., 2010. The Pascal Visual Object Classes (VOC) Challenge. *International Journal of Computer Vision*, 88(2), 303–338. <https://doi.org/10.1007/s11263-009-0275-4>.
- Geirhos, R., Rubisch, P., Michaelis, C., Bethge, M., Wichmann, F. A., Brendel, W., 2019. ImageNet-trained CNNs are biased towards texture; increasing shape bias improves accuracy and robustness. *arXiv:1811.12231 [cs, q-bio, stat]*. <http://arxiv.org/abs/1811.12231>. arXiv: 1811.12231.

- Ghorbanzadeh, O., Tiede, D., Dabiri, Z., Sudmanns, M., Lang, S., 2018. DWELLING EXTRACTION IN REFUGEE CAMPS USING CNN – FIRST EXPERIENCES AND LESSONS LEARNED. *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLII-1, 161–166. <https://www.int-arch-photogramm-remote-sens-spatial-inf-sci.net/XLII-1/161/2018/>.
- Girshick, R., Radosavovic, I., Gkioxari, G., Dollár, P., He, K., 2018. Detectron. <https://github.com/facebookresearch/detectron> (20 April 2020).
- He, K., Gkioxari, G., Dollár, P., Girshick, R., 2017. Mask R-CNN. *Proceedings of the IEEE International Conference on Computer Vision*, 2961–2969.
- He, K., Zhang, X., Ren, S., Sun, J., 2016. Deep residual learning for image recognition. *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770–778.
- Katz, I., 2016. A network of camps on the way to Europe. *Forced Migration Review*, 51, 17–19.
- LeCun, Y., Bengio, Y., Hinton, G., 2015. Deep learning. *Nature*, 521(7553), 436–444. <http://www.nature.com/articles/nature14539>.
- Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C. L., 2014. Microsoft COCO: Common Objects in Context. *Computer Vision – ECCV 2014*, 8693, Springer International Publishing, Cham, 740–755.
- Mikołajczyk, A., Grochowski, M., 2018. Data augmentation for improving deep learning in image classification problem. *2018 international interdisciplinary PhD workshop (IIPhDW)*, IEEE, 117–122.
- Quinn, J. A., Nyhan, M. M., Navarro, C., Coluccia, D., Bromley, L., Luengo-Oroz, M., 2018. Humanitarian applications of machine learning with remote-sensing data: review and case study in refugee settlement mapping. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 376(2128), 20170363. <http://rsta.royalsocietypublishing.org/lookup/doi/10.1098/rsta.2017.0363>.
- Rankin, J., Smith, H., Connolly, K., McKernan, B., 2020. Refugees told 'Europe is closed' as tensions rise at Greece-Turkey border. <https://www.theguardian.com/world/2020/mar/06/refugees-europe-closed-tensions-greece-turkey-border> (20 April 2020).
- Ren, S., He, K., Girshick, R., Sun, J., 2017. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(6), 1137–1149. <http://ieeexplore.ieee.org/document/7485869/>.
- Spröhnle, K., Tiede, D., Schoepfer, E., Füreder, P., Svanberg, A., Rost, T., 2014. Earth Observation-Based Dwelling Detection Approaches in a Highly Complex Refugee Camp Environment — A Comparative Study. *Remote Sensing*, 6(10), 9277–9297. <http://www.mdpi.com/2072-4292/6/10/9277>.
- Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R., 2014. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1), 1929–1958.
- UNHCR, 2015. UNHCR operations management cycle. <https://emergency.unhcr.org/entry/52635/unhcr-operations-management-cycle> (20 April 2020).
- Wickert, L., Bogen, M., Richter, M., 2020. Dwelling Detection on VHR satellite imagery of Refugee camps using a Faster R-CNN. *Proceedings of the 2nd International Conference on Advances in Signal Processing and Artificial Intelligence*, IFSA Publishing, Berlin.
- Yao, Y., Rosasco, L., Caponnetto, A., 2007. On Early Stopping in Gradient Descent Learning. *Constructive Approximation*, 26(2), 289–315. <https://doi.org/10.1007/s00365-006-0663-2>.
- Zhang, L., Zhang, L., Du, B., 2016. Deep Learning for Remote Sensing Data: A Technical Tutorial on the State of the Art. *IEEE Geoscience and Remote Sensing Magazine*, 4(2), 22–40. <http://ieeexplore.ieee.org/document/7486259/>.
- Zhao, Z.-Q., Zheng, P., Xu, S.-T., Wu, X., 2019. Object Detection With Deep Learning: A Review. *IEEE Transactions on Neural Networks and Learning Systems*, 1–21. <https://ieeexplore.ieee.org/document/8627998/>.
- Zhu, H., Meng, F., Cai, J., Lu, S., 2016. Beyond pixels: A comprehensive survey from bottom-up to semantic image segmentation and cosegmentation. *Journal of Visual Communication and Image Representation*, 34, 12–27. <https://linkinghub.elsevier.com/retrieve/pii/S1047320315002035>.