Complete Model for Automatic Object Detection and Localisation on Aerial Images using Convolutional Neural Networks

Dunja Božić-Štulić, Stanko Kružić, Sven Gotovac, and Vladan Papić

***Abstract*—In this paper, a novel approach for an automatic object detection and localisation on aerial images is proposed. Proposed model does not use ground control points (GCPs) and consists of three major phases. In the first phase, optimal flight route is planned in order to capture the area of interest and aerial images are acquired using unmanned aerial vehicle (UAV), followed by creating a mosaic of collected images to obtained larger field-of-view panoramic image of the area of interest and using the obtained image mosaic to create georeferenced map. The image mosaic is then also used to detect objects of interest using the approach based on convolutional neural networks.**

***Index Terms*—georeferencing, GIS, UAV, image mosaic, object detection, convoloutional neural networks**

I. Introduction

Aerial images are widely used in various activities by providing visual records. This type of remotely sensed images is helpful in generating digital maps, managing ecology, monitoring crop growth, region surveying, etc. Also, it can be a helpful aid in search and rescue operations. Conventional aerial photographs are an essential source of data for natural resource scientists.

High-quality aerial imagery can be acquired using conventional platforms such as satellites and aircraft but their temporal resolution is limited by the restricted availability of aircraft platforms and orbit characteristics of satellites [1]. This limits their use for map updating purposes, as it increases costs and production time. Recently, UAVs have been introduced in mapping activities and have been linked with the low-cost production of accurate and high-quality spatial data in a short time [2]. Several approaches for UAV-based georeferencing have been proposed recently. In [3], global position system (GPS) information was used to provide the coordinates of the aerial photo centre, but variations in pitch and roll were not catered for, thereby restricting the UAV altitude. In [4], GPS information was used, but sequential triangulation was also needed for updating the camera parameters. A robust image matching procedure had then to be applied in real time for finding tie points. Xiang and Tian [5] proposed a method for georeferencing, but the orientation and position of the camera with respect to the UAV was not generic. In [6], the proposed model for automatic georeferencing of images obtained by UAV, camera position and orientation with respect to the UAV are not restricted. Hence, no simplification is possible in the pixel mapping, and the pixel positions necessary for camera calibration are obtained entirely by image processing, i.e., automatically. In [7], the mapping model was proposed, but GCPs were necessary for the implementation of the model.

Object detection is common task in computer vision, used for autonomous vehicles, smart video surveillance, facial detection and various people counting applications. Systems like this are not only used for recognizing and classifying every object in an image, but also for localizing each one by drawing the appropriate bounding box around it. Several approaches for detection and localization of objects using CNNs have been proposed. Radovic *et al* [8] have tested CNN - based software called "YOLO" for object recognition in satellite images. They managed to get 97.5% accuracy, but for they validation phase they used satellite images. In [9] authors used R-CNN for car detection in aerial images. Authors' main focus was on detection of small objects on aerial images.

The main contributions of this paper are the following:

1) the procedure for creation of the route path of UAV is proposed;
2) a technique for creating mosaic of images acquired by UAV and loading the mosaic with the appropriate world file transformation in GIS software is proposed and implemented;

The paper is structured as follows. In Section 2, the UAV unit and flight route planning are presented, and procedures for image mosaicking, georeferencing and object detection are implemented and explained. In Section 3, research results are presented, while in Section 4 conclusions are drawn and directions for future work are given.

II. Proposed Model

Our proposed model for automatic object detection and localisation of aerial images consists of three major phases. The first phase is data acquisition using UgCS software for flight planning and DJI Phantom 3 Professional UAV. After data acquisition, next phase is creating a mosaic of collected images in order to obtain a larger field-of-view panoramic image of the area of interest and computing the world file in

Manuscript received January 16, 2018; revised March 5, 2018. Date of publication: March 15, 2018.

D. Božić-Štulić, S. Kružić, S. Gotovac and V. Papić are with the University of Split, Faculty of Electrical Engineering, Mechanical Engineering and Naval Architecture, Split, Croatia.
E-mails:{dgotovac, skruzic, gotovac, vpapic}@fesb.hr

Digital Object Identifier (DOI): 10.24138/jcomss.v14i1.441

order to properly georeference obtained panoramic image onto a map. Finally, in the last phase, image mosaic is divided into smaller pieces which were used as inputs to the convolutional neural network (CNN) for detection of the objects of interest.

A. Image acquisition phase

1) UAV Unit: The UAV unit used for image acquisition in the study was DJI Phantom 3 Professional [10], a quadrotor UAV which features 3-axis stabilisation using gimbal, vertical accuracy of up to 0.5 m, and horizontal accuracy of up to 1.5m using GPS (which can be further more accurate at lower speeds and altitudes, by switching to vision-based positioning, using both ultrasound and image data). It also features a built-in camera for which specifications are presented in Table I.

TABLE I
SPECIFICATIONS OF CAMERA USED IN THE RESEARCH

Model	DJI Phantom 3 Professional
Sensor	Sony EXMOR 1/2.3"
Lens	FOV 94° 20mm (35mm format equiv.) f/2.8
Effective resolution	12.4 MP
Sensor width and height	6.48525 4.86394
Image size	4000×3000
Video size	4096×2160 @ 24fps; 4K @ 30fps
Pixel size	1.4
Focal length	5

2) Flight route planning: To successfully capture the area of interest, a good flight route plan is a fundamental requirement. Since manually flying a drone to follow a flight route is very difficult, the appropriate software has to be used. In the research, we used software UgCS (Universal Ground Control Station) [11] for our flight route planning. The software can compute the route and fly the UAV autonomously. Also, the appropriate input parameters need to be set accordingly: the area of interest must be marked on a map, adjacent image overlap percentage and camera properties must be set (see Fig. 2a). These input parameters are used for calculation of the optimal flight route which will assure full coverage of the area of interest. More details about flight route planning are shown in the following list, while the example of the computed flight route may be seen in Fig. 2b.

Flight route planning process consists of seven steps:

1) choose location on Google Maps (as required by UgCS software);
2) set UAV type;
3) set the home location for UAV;
4) set flight properties (side and forward overlap between adjacent images, flight altitude);
5) calculate optimal route path using user-provided area of interest given flight properties previously set;
6) upload route to the UAV unit;
7) start the planned flight;

During route planning, GCPs were labelled, which means that metadata about every image that was taken was available. Also, geolocation data (latitude, longitude, altitude), field of view (FOV) and resolution were all available. The problem is that all those data depart from the values in the real world. The algorithm that was used for obtaining world file is based on obtaining a mosaic of a taken set of photographs using metadata and correction offset by the algorithm for image processing.

B. Image mosaicking

Image mosaicking is a process where image processing techniques are applied to a set of aerial images in order to create larger field-of-view panoramic images that are impossible to capture in a single image. The process is similar to well-known image stitching, the only difference being one that in image mosaicking only aerial images are used, which results in less intensive transformation calculations because the compositing surface is the approximately planar. The process consists of several steps which are described in following paragraphs.

1) Feature detection and description: Feature detection is a process of finding key points in images, while feature description is a process of extraction of local image patch around detected features in order to "describe" them. There exist numerous methods for feature detection and description: Harris, Scale-Invariant Feature Transform (SIFT), Speeded Up Robust Features (SURF), Features from Accelerated Segment Test (FAST) and Oriented FAST and Rotated BRIEF (ORB), to name a few that are most widely used in computer vision applications.

In this paper, SIFT was the method of our choice for feature detection and description and is explained thoroughly in the following paragraphs. SIFT is an algorithm for both feature detection and description developed and published by David G. Lowe in 1999 [12], and further enhanced in 2004 [6]. SIFT feature descriptor is invariant to orientation, scale and partially to affine transform.

SIFT keypoints are identified as local extrema of Difference of Gaussians (DoG)

$$D(\mathbf{x}, \sigma_1, \sigma_2) = I(\mathbf{x}) * (G(\mathbf{x}, \sigma_1) - G(\mathbf{x}, \sigma_2))$$

where $\mathbf{x} = [\ x \quad y\ ]^T$, $I(\mathbf{x})$ is original image, and $G(\mathbf{x}, \sigma)$ is Gaussian blur, applied in scale space to Gaussian blurred and resampled images. Each pixel of DoG image is thresholded with 8 neighbouring pixels on the same scale, as well as with 9 appropriate pixels in neighbouring scales. If a pixel is local extrema found by described procedure, it is a potential keypoint. This procedure is repeated multiple times for each octave in the Gaussian pyramid. Once all potential keypoints are found, they must be refined to obtain more accurate results. The Taylor series expansion of scale space

$$D(\mathbf{x}) = D + \frac{\partial D^T}{\partial \mathbf{x}}\mathbf{x} + \frac{1}{2}\mathbf{x}^T\frac{\partial^2 D}{\partial \mathbf{x}^2}\mathbf{x}$$

is used to obtain more accurate location of extrema. If an intensity at extrema is less than a threshold value, they are rejected. In our research, value of 0.03 was used, as per [6].

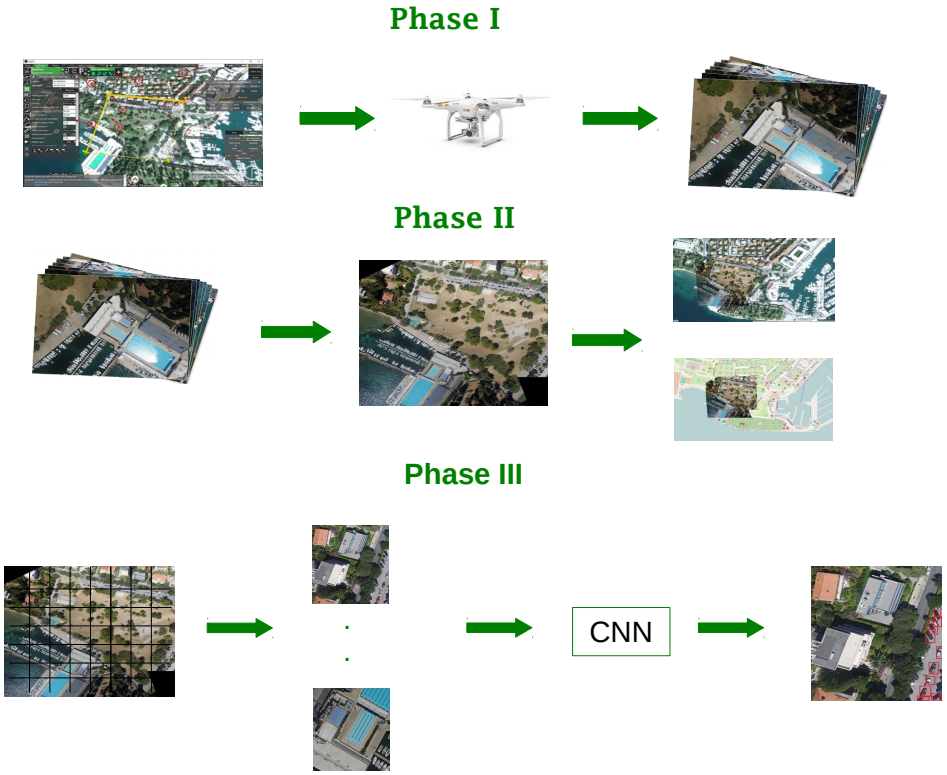
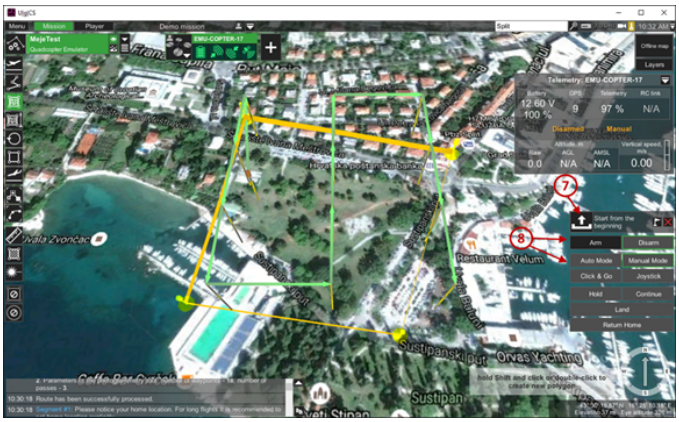


Fig. 1. Proposed model for automatic object detection and localisation on aerial images. In the first phase, the flight plan is created and data is acquired using UAV unit, followed by creating a mosaic of acquired images and georeferencing it on a map using GIS software. Finally, image mosaic is divided into smaller chunks to perform object detection using CNN.

(a) User marks corners of the study area on the map



(b) Green arrows representing optimal flying route needed to capture desired study area

Fig. 2. UgCS software for flight route planning

Since DoG has strong responses along edges, so they must be removed. Hessian matrix is used for that purpose that is computed at the scale and location of the edge keypoint:

$$H = \begin{bmatrix} D_{xx} & D_{xy} \\ D_{xy} & D_{yy} \end{bmatrix}$$

The ratio of eigenvalues of H is taken and thresholded against a constant value of $r = 10$ [6] which is called edge threshold. If the ratio is greater than edge threshold the point is discarded.

When low-contrast keypoints, as well as edge keypoints are discarded, what is left are the keypoints that are later used in further steps of image mosaicking process.

SIFT keypoint descriptor is computed as follows. A set of orientation histograms is created on 4x4 pixel region with 8 bins each. Histograms are computed from gradient magnitude and orientation values of samples in a 16 x 16 region around the keypoint such that each histogram contains samples from a 4x4 subregion of the original region. The descriptor then becomes a vector of all the values of these histograms and has 128 elements, since there are 16 histograms each with 8 bins. The descriptor is then normalised to unit length in order to make it (at least partially) invariant to affine transformation.

2) Feature matching: The objective of the feature matching stage is to find correspondences between overlapping images. Those correspondences can be found in many different ways. The simplest one is a pairwise comparison of images' feature descriptors. There exist a number of feature matching techniques, the most common ones used being exhaustive search and nearest neighbour techniques. The distance measures used are $L2$ (Euclidean) distance for vector feature descriptors.

For purpose of this research, feature matching can be further optimised. Considering flight plan described in previous sections, input images are given in sequence and are organised in a grid-like structure, approximate relationship between each image pair is known. Originally, feature matching step's execution time is proportional to squared number of input images. However, if only a smaller subset of images, those that are adjacent, are matched pairwise, a huge amount of computation is avoided with negligible loss. For example, consider an image not on the edge of the grid-like structure. That image has eight other overlapping images (one to the each of left, right, up, down edges, and to the each of upper-left, upper-right, lower-left and lower-right corners of the image). If the image is on the edge of grid-like structure, number of other overlapping images if even smaller. This is done by using a $N \times N$ matrix which represents binary mask (N being number of input images). That matrix value is set to 1 if images at appropriate row and column are adjacent and 0 otherwise. During the feature matching stage, only image pairs that have binary mask set to 1 are evaluated and pairwise matched.

Once correspondences are found, there is a need to find a subset of those correspondences which produce accurate alignment (inliers) of the images and that is consistent with a particular camera motion estimate. This is done by using Random Sample Consensus (RANSAC) algorithm. RANSAC [13] is an iterative algorithm used for fitting a model to observed data which contain outliers. Since the motion model for aerial images is affine, which has 6 parameters, we need at least three points correspondences to estimate the affine model. RANSAC starts by selecting a subset of three point correspondences at random and calculates the affine model with them. Then, all other correspondences are examined if they are located within a tolerance of their location predicted by the calculated model. If the ratio of the number of inlier correspondences to the total number of correspondences is greater than a predefined threshold, a model is re-estimated with all inliers. Then, the whole process is repeated, a maximum of M times, where M is big enough to ensure a high probability that random subset does not contain an outlier. A model with the largest number of inliers is kept as final.

3) Image blending: Once feature matching is finished, images are warped together on a compositing surface. As images are aerial and considering that ground is approximately flat when altitude is large enough, a planar compositing surface is chosen.

Once pixels from source images have been mapped onto the composite surface, there may arise the need for blending in order to create an attractive-looking panorama. If all of the images are in perfect alignment, there is no need for blending. However, visible seams (due to exposure differences), blurring (due to misalignment), or ghosting (due to moving objects) often occur in real images. Creating clean panoramas involves both deciding which pixels to use and/or how to weight or blend them. Feathering (weighting), simple blending method will be briefly described in following lines. It takes values of each pixel in a blended (overlapping) region and computes average:

$$C(\mathbf{x}) = \frac{\sum_k w_k(\mathbf{x}) \tilde{I}_k(\mathbf{x})}{\sum_k w_k(\mathbf{x})}$$

where $\tilde{I}_k(\mathbf{x})$ are warped images and $w_k(\mathbf{x})$ is weighting functions. As per [14], good choice for weighting functions are ones that weights pixels in the centre of the image more heavily than those near edges.

Image blending stage is optional. Since there are uses when there is no need for attractive-looking image mosaics (e.g. in search and rescue operations) it is avoided in order to save time.

Using all previously described methods, a procedure for creation of the aerial mosaics is given in 1.

Algorithm 1 Mosaicking of aerial images

Input: Sequence of N images
  Extract features from all N aerial images
  for all images **do**
    Match feature pairs between adjacent images
    Find geometrically consistent transformation using feature matches and RANSAC
  Warp images to compositing surface using estimated transforms
  Blend resulting mosaic *(optional)*
Output: Image mosaic

TABLE II
WORLD FILE VALUES

Pixel size in the x-direction in map units/pixel	0.037633283681665
Rotation about y-axis	0
Rotation about x-axis	0
Pixel size in y-direction in map units	-0.039545625330468
x-coordinate of the center of the upper left pixel	1828175.26
y-coordinate of the center of the upper left pixel	5389028.73

C. World file transformation

After creation of the mosaic, values needed for world file transformation can easily be computed. In general, world files use the same name as the image, with a "w" appended. For example, the world file for the image file split.tif would be called split.tifw, and the world file for split1.rlc would be split1.rlcw. However, since GIS only accepts 3-letter file extensions, the first and third characters of the image file's suffix and a final "w" are used for the world file suffix. Therefore the world files for split.tif and split1.rlc would be split.tfw and split1.rcw, respectively. Table II shows world file transformation values that were computed for our mosaic image created with previously describe technique.

D. Georeferencing

The resulting image mosaic with applied world file transformation has to be processed in QGIS [15], basemap layers for were chosen, and georeferenced image mosaic was overlaid on top of them. Since QGIS is an open source GIS software and has a possibility of adding basemap layers via a plug-in, two base map layers were chosen for this experiment. First is Open Street Map and second is Google Satellite (Fig. 4). However, more basemap layers are also available for usage via plugins.

Georeferencing is the process of aligning a raster dataset to known map coordinates and assigning a coordinate system. It creates additional information within the file itself and/or in supplementary files that accompany the image file that tells GIS software how to properly place and draw it. It is a crucial step for making aerial and satellite imagery useful for mapping [7].

Different maps may use different projection systems. There are three known projections: cylindrical, conical and azimuthal (planar). A cylindrical projection is analogous to wrapping a cylinder of paper around the Earth, projecting the Earths features onto it, and then unwrapping the cylinder. A conical projection is analogous to wrapping a sheet of paper around the Earth in a cone. An azimuthal or planar projection is analogous to touching the Earth with a sheet of flat paper. Any projection will distort the Earth in some way.

Georeferencing tools contain methods to combine and overlay these maps with minimum distortion. The conformal property when shapes of small features are preserved, or in other words, scales of the projections in x and y directions are always equal and equal-area property when shapes are distorted, but areas measured on the map are always in the same proportion to areas on the Earths surface [16]].

Using georeferencing methods, data obtained from an observation or surveying may be given a point of reference from topographic maps already available. In the case of projecting the earths curved surface on a flat surface, distortion of one or more features will occur. The conventions for locating points on the Earths surface for purposes of nautical and aeronautical navigation (long distances on small scale charts) is generally best conducted using latitude and longitude (spherical coordinates). Locating points on large-scale maps and for ground navigation is generally best accomplished with Cartesian-style plane coordinates. Large-scale maps can treat the Earths surface as a plane, taking the advantage of that simple geometric shape and mathematics rather than a complex sphere. Properly constructed large-scale maps, such as topographic maps, take the curvature of the Earth into account. Simple linear increments (i.e. meters) of plane coordinates are significantly easier for large-scale map users to handle accurately at high precision in the field than the more complex angular increments of latitude and longitude (i.e. degrees).

In general, the georeferencing process consists of the following steps:

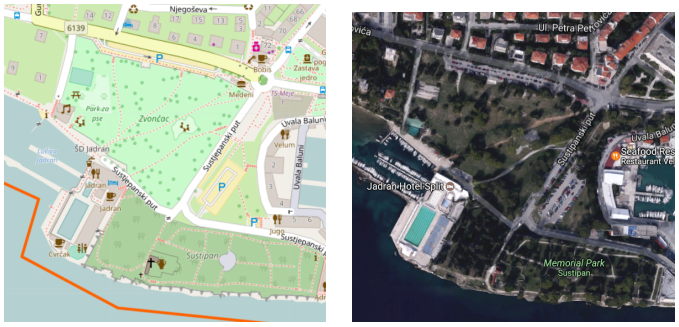
1) Identification of appropriate reference data. A georeferenced dataset is needed in the desired coordinate system (preferably the same as a scanned map or digital image) which will be used to align with and the raster data (target data). The raster data and the reference data have to have some features in common that are visible in both datasets, such as street intersections, hydrographic features or building outlines.
2) A selection of control points (based on common features) to link known locations in both datasets.
3) Transformation of the target data to align with the reference data.

E. Object detection and localization

Deep CNNs are composed of several layers of processing each containing linear as well as nonlinear operators which are jointly learnt in an end-to-end way to solve specific tasks [17], [18]. Specifically, deep CNNs are commonly made up of convolutional, normalization, pooling, and fully connected layers. The convolutional layer is the main building block of the CNN, and its parameters consist of a set of learnable filters. Each filter is spatially small (along width and height), but extends through the full depth of the input image. The feature maps produced via convolving these filters across the input image are then fed into a non-linear gating function such as the rectified linear unit (ReLU) [19]. Next, the output of this activation function can be further subjected to normalization (i.e., local response normalization) to help in generalization. The pooling layer takes small rectangular blocks from the convolutional layer and subsamples it to produce a single output from each block. The literature conveys several ways to perform pooling, such as taking the average, the maximum, or a learned linear combination of the values in the block. This layer allows control of over-fitting and reduces the amount of parameters and computations.

For object detection and localization we used pre-trained

Fig. 3. Image mosaic of the area of interest before (left) and after (right) applying world file transformation



(a) Open Street Map (b) Google Satellite

Fig. 4. Basemap layers examples

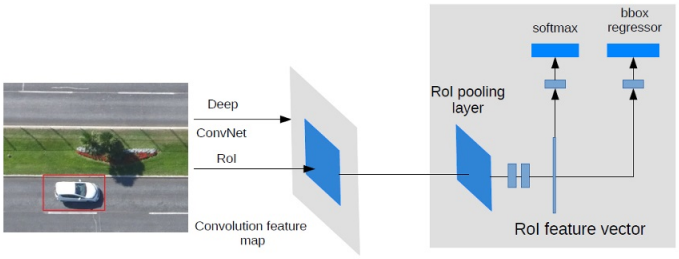


Fig. 5. Architecture of pre-trained RCNN network. Reproduced from [21].

Faster R-CNN model [20]. Faster R-CNN is composed of two modules, first module is deep fully connected convolution network that proposes regions, and the second one is Fast R-CNN detector [21]. Using the recently popular terminology of neural networks with 'attention' [22] mechanisms, the Region Proposal Network (RPN) tells the Fast R-CNN module where to look. The RPN module takes an image as input and proposes a set of rectangular objects. Figure 5 shows architecture of RCNN network, which we used for car detection.

1) Dataset preparation: For training pre-trained RCNN we created training dataset. Training dataset contains 200 images. Initial dataset is collected on the different location from our testing flight and contained 60 images in resolution of 3000x4000. Since for input image we needed 333x500 resolution, we divided original image into parts (only those were cars are present). With this approach we managed to get 200 images. Figure 6. shows example images from training dataset. Collected images were manually labelled with bounding boxes of cars present on each of them.

Fig. 6. Example images from training dataset

TABLE III
DETECTION RESULTS

Image	Car present	TP	FP	Pacc	Uacc
Image 1	61	49	3	80.32%	83.6%
Image 2	27	20	2	74.07%	81.48%

2) Detection results: To assess the capability of our methodology for correct identification, we considered the accuracy measures of the producers and users, where TP are the true positives (i.e., the number of cars correctly identified), FP are the false positives (i.e., the number of cars incorrectly identified), and N is the real number of cars present in the image. Figure 7. shows detection and localization results.

Fig. 7. Detection and localization results on test images

III. RESULTS

In the research, 125 aerial images of the study area (Split, Croatia) were acquired with the UAV described in section II-A1. Fig. 1 depicts proposed model used in the research. The first step in the experiment was to create a proper flight plan with the desired side and forward overlap. For the creation of flight plan, UgCS software was used. Setting appropriate overlap between the images is very important, because it may result in an inadequate number of features when matching between the images when the overlap is too small. Appropriate overlaps are at least 40%, while 60% overlap was used in the experiment. The user needs to specify corner points on the map of the area of interest, and the software will calculate the optimal flight route.

Image orientation with geotags which, for this case, resulted in low accuracy geolocation, especially regarding height component. The reason for this inaccuracy is the use of the onboard GPS of the UAV. However, obtained results are promising and can be used for mapping applications which require less than 92 cm of accuracy. Choice of the reference layer usually depends on the application type: monitoring crop growth, region surveying, search and rescue, etc. In the case of planning the search and rescue operations, the different referenced layer is needed. In this study, we mapped acquired images on two different layers. The first layer was Google Satellite layer which can be used for monitoring changes in the area. The second one was Open Street Map and this layer was used to see the actual state on the field. Therefore, the reference image is chosen, usually an image containing many different objects that can be used as GCPs. After all aforementioned steps, image georeferencing is started using TPS transformation [7] that remove distortions from the image. Finally, the image is ready for mapping.

The georeferencing process was completed using our own implementation of world file calculator, described in section II-C ,which was then applied to image mosaic obtained using the procedure described in section II-B. Our proposed model model provides a map with good quality and visibility of its features with objects easily detectable. However, some minor deformations were detected in the study area. Those include facade visibility, moving objects and still objects. However, errors were very small and hardly detectable. Also, it is obvious that there are differences between two base map layers. Examples of georeferenced image mosaics overlaid on top of two base map layers (namely, Open Street Map and Google Satellite) are shown in Fig. 8.

Our system consists of a ground-based computer, where all processing is done, and an UAV unit, which is used just for taking photographs. The system was tested with aerial images taken using DJI Phantom 3 Professional, with 4000×3000 px in size and JPG compressed. For mission planning, UgCS software was used, where it was possible to configure various details about flight and camera of the UAV unit, the most important being percentage of overlap between the images in each direction and flight velocity. The user needs to specify corner points on the map of the area of interest, and the software will calculate the optimal flight route. The UAV camera can be configured to take images with constant exposure and focus which is very important in this method because it makes blending step optional while keeping image usable and nice-looking without visible edges.

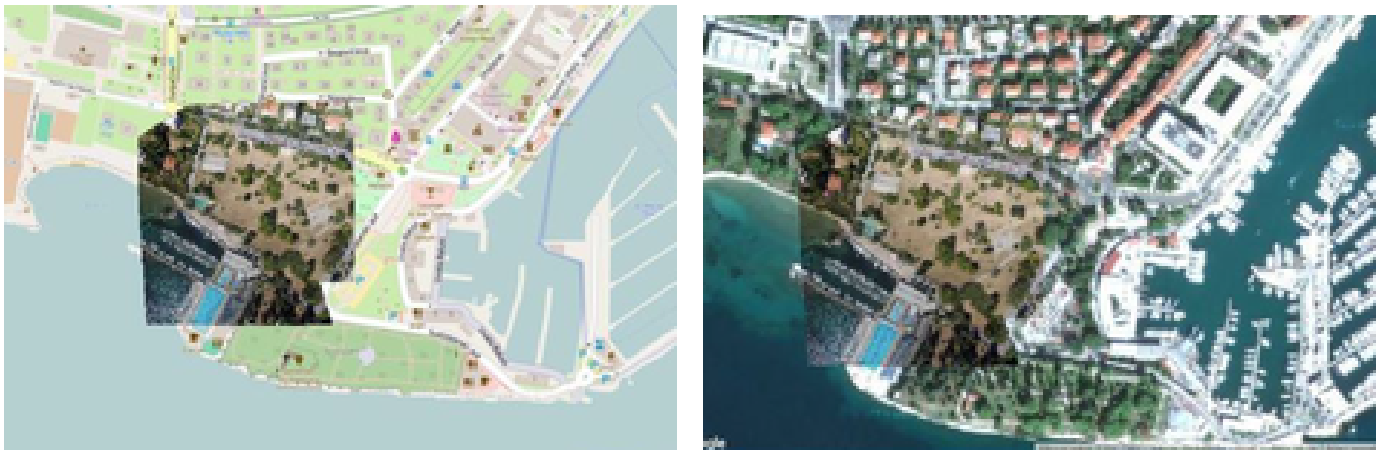
This result shows that proposed model for automatic image georeferencing has shown great performances for search and rescue purposes. Objects can be easily detected, which is most important, also the changes in terrain can be detected in short time.

IV. CONCLUSION

A novel approach for automatic object detection and localisation on aerial images that were taken using UAVs was proposed in the paper. The main idea of our approach was to create an optimal flight route plan to capture desired area, make a mosaic of collected images, create world file transformation and load the mosaic image with the appropriate world file in GIS software, as well as to detect objects of interest and their locations in the resulting image and on the map (i.e. their real-world geographic coordinates).

The results indicate that, by using UAVs, with proper training and with applying adequate techniques, it is possible to obtain high-quality photogrammetric products comparable to ground surveying equipment. Comparing to the time and costs it would have taken to produce such data using traditional equipment, UAVs are a more promising alternative for photogrammetric surveying.

However, the obtained quality of UAV photogrammetric products depends on many elements which needed to be taken care of at every step. The flight route planning needs to be prepared properly so that information about the percentage



(a) Open Street Map

(b) Google Satellite

Fig. 8. Image mosaic mapped and overlaid on top of basemap layers

of side and forward overlap between the images of the study area is set. Input images for proposed system are given in sequence and they are organised in a grid-like structure, so each image overlaps with only a small number of others. Also, an approximate relationship between the images is known due to the grid-like organisation and amount of overlap set. Additionally, there is usually no need for blending the final image mosaic, since images are in most use cases taken with the constant exposure and focus.

The final map, which is a product of mapping the mosaic image from UAV has some errors. These deformations were caused by lack of images or overlap during image acquisition, hence the chosen ground control points were not dense enough to perform the geometric reconstruction of objects.

However, these deformations did not have much impact in this work and some of them were easily removed using QGIS plugins. The first step of flight route planning and image acquisition needs to be done accurately so that the final result will be high quality. This novel approach has shown great performances in monitoring terrain abnormalities since the terrain is sustainable to the weather changes and time of the year. It can provide a help to search and rescue teams in their operations since object detection in a high-quality mosaic image is generally not a difficult problem.

REFERENCES

[1] Z. Zhang, "A flexible new technique for camera calibration," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 22, no. 11, pp. 1330–1334, 2000.

[2] H.-J. Song, Y.-Z. Chen, and Y.-Y. Gao, "Velocity calculation by automatic camera calibration based on homogenous fog weather condition," *International Journal of Automation and Computing*, vol. 10, no. 2, pp. 143–156, 2013.

[3] J. Mark and P. Hardin, "Applications of inexpensive remotely piloted vehicles (rpvs) for collection of high resolution digital imagery," in *Proceedings of the 20th Biennial Workshop on Aerial Photography, Videography, and High-resolution Digital Imagery for Resource Assessment*, 2005.

[4] K. Choi and I. Lee, "Real-time georeferencing of image sequence acquired by a uav multi-sensor system," in *Multi-Platform/Multi-Sensor Remote Sensing and Mapping (M2RSM), 2011 International Workshop on*. IEEE, 2011, pp. 1–6.

[5] H. Xiang and L. Tian, "Method for automatic georeferencing aerial remote sensing (rs) images from an unmanned aerial vehicle (uav) platform," *Biosystems Engineering*, vol. 108, no. 2, pp. 104–113, 2011.

[6] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International journal of computer vision*, vol. 60, no. 2, pp. 91–110, 2004.

[7] D. Gotovac, S. Gotovac, and V. Papić, "Mapping aerial images from UAV," in *Computer and Energy Science (SpliTech), International Multidisciplinary Conference on*. IEEE, 2016, pp. 1–6.

[8] M. Radovic, O. Adarkwa, and Q. Wang, "Object recognition in aerial images using convolutional neural networks," *Journal of Imaging*, vol. 3, no. 2, p. 21, 2017.

[9] J. B. Lars W. Sommer, Tobias Schuchert, "Deep learning based multi-category object detection in aerial images," pp. 10 202 – 10 202 – 8, 2017. [Online]. Available: https://doi.org/10.1117/12.2262083

[10] Phantom 3 Professional. [Online]. Available: https://www.dji.com/phantom-3-pro

[11] Ground Station Software — UgCS PC Mission Planning. [Online]. Available: https://www.ugcs.com/

[12] D. G. Lowe, "Object recognition from local scale-invariant features," in *Computer vision, 1999. The proceedings of the seventh IEEE international conference on*, vol. 2. Ieee, 1999, pp. 1150–1157.

[13] M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, vol. 24, no. 6, pp. 381–395, 1981.

[14] R. Szeliski, "Image alignment and stitching: A tutorial," Tech. Rep., October 2004. [Online]. Available: https://www.microsoft.com/en-us/research/publication/image-alignment-and-stitching-a-tutorial/

[15] QGIS Project. [Online]. Available: http://www.qgis.org/en/site/

[16] J. Wieczorek, Q. Guo, and R. Hijmans, "The point-radius method for georeferencing locality descriptions and calculating associated uncertainty," *International journal of geographical information science*, vol. 18, no. 8, pp. 745–767, 2004.

[17] Y. LeCun, C. Farabet, C. Couprie, and L. Najman, "Learning hierarchical features for scene labeling," *IEEE Transactions on Pattern Analysis, Machine Intelligence*, vol. 35, pp. 1915–1929, 08 2013. [Online]. Available: doi.ieeecomputersociety.org/10.1109/TPAMI.2012.231

[18] T. Brosch and R. Tam, "Efficient training of convolutional deep belief networks in the frequency domain for application to high-resolution 2d and 3d images," *Neural Comput.*, vol. 27, no. 1, pp. 211–227, Jan. 2015.

[19] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems 25*, F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, Eds. Curran Associates, Inc., 2012, pp. 1097–1105. [Online]. Available: http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf

[20] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Advances in Neural Information Processing Systems 28*, C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, Eds. Curran Associates, Inc., 2015, pp. 91–

99. [Online]. Available: http://papers.nips.cc/paper/5638-faster-r-cnn-towards-real-time-object-detection-with-region-proposal-networks.pdf
[21] R. Girshick, "Fast r-cnn," in *Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV)*, ser. ICCV '15. Washington, DC, USA: IEEE Computer Society, 2015, pp. 1440–1448. [Online]. Available: http://dx.doi.org/10.1109/ICCV.2015.169
[22] J. Chorowski, D. Bahdanau, D. Serdyuk, K. Cho, and Y. Bengio, "Attention-based models for speech recognition," *CoRR*, vol. abs/1506.07503, 2015. [Online]. Available: http://arxiv.org/abs/1506.07503

Dunja Božić-Štulić was born 16.04.1990. She received her Master of Engineering degree in Electronics and Computer engineering in 2014, from Faculty of Electrical Engineering, Mechanical Engineering and Naval Architecture, University of Split, Croatia. She is currently working as research assistant at Faculty of Electrical Engineering, Mechanical Engineering and Naval Architecture, University of Split, Croatia. Her research interests include machine learning, deep learning and artificial intelligence.

Stanko Kružić was born 11th August 1985. He received his Master of Engineering degree in Automation and Systems in 2009, from Faculty of Electrical Engineering, Mechanical Engineering and Naval Architecture, University of Split, Croatia. He is currently working as research assistant at Faculty of Electrical Engineering, Mechanical Engineering and Naval Architecture, University of Split, Croatia and is PhD candidate. The fields of primary scientific interest are mobile robotics, remote sensing, image processing and artificial intelligence.

Sven Gotovac was born on 07.22.1960. He graduated in 1983. at the University of Split, Faculty of Electrical Engineering, Mechanical Engineering and Naval Architecture. He received master's degree at the University of Zagreb, Faculty of Electrical Engineering in 1988 and his PhD at the TU Berlin in 1994.From 1884 he worked at the University of Split, Faculty of Electrical Engineering, Mechanical Engineering, and Naval Architecture. Currently he is full professor and lead of the Department of Computer Architecture and Operating Systems. He worked on the three national research projects, one international, and has been leader of two national and currently leader of one international project at the ALICE experiment in CERN. Currently he is the Dean of the Faculty of electrical engineering, mechanical engineering and naval architecture, University of Split. He is co-author on about a hundred scientific papers in indexed journals, 15 papers at international conferences and co-author of two books. He was mastering four PhDs and five master's theses(http://bib.irb.hr/lista-radova?autor=108173). He is married, father of four children. He speaks English, German and Italian.

Vladan Papić received the B.Sc. degree from the Faculty of Electrical Engineering, Mechanical Engineering and Naval Architecture, University of Split, Croatia in 1993, and the M.Sc. and Ph.D. degrees from the same university in 1996 and 2002, respectively. He is a Professor with the University of Split, Croatia and the Vice-dean for Bussines activities. His research interests include computer vision and intelligent systems.