Multiview Vision-based Human Crowd Localization for UAV Fleet Flight Safety

Efstratios Kakaletsis, Ioannis Mademlis, Nikos Nikolaidis, Ioannis Pitas

Department of Informatics, Aristotle University of Thessaloniki
Thessaloniki, Greece
Emails:{ekakalets, imademlis, nnik, pitas}@csd.auth.gr

Abstract

This paper presents a centralized, vision-based method for robust, on-the-fly 3D localization and mapping of human crowds in large-scale outdoor environments, assuming their independent visual detection on the camera feed of multiple UAVs. The proposed method aims at enhancing vision-assisted human crowd avoidance, in line with common UAV safety regulations, since the resulting 3D crowd annotations may be employed by other algorithms for on-line mission/path replanning during deployment of a UAV fleet. Initially, 2D crowd heatmaps are assumed to be derived per video frame on-board each UAV separately, using deep neural human crowd detectors, which indicate the probability of each pixel depicting a human crowd. The UAV-mounted cameras are assumed to be covering the same large-scale outdoor area over time. The heatmaps of each time instance are transmitted to a central computer and back-projected onto the common 3D terrain/map of the navigation environment, utilizing the intrinsic and extrinsic camera parameters. The projected crowd heatmaps derived from the different drones/cameras are fused by exploiting a Bayesian filtering approach that favors newer crowd observations over older ones. Thus, during flight, an area is marked as crowded (therefore, a no-fly zone) if all, or most, individual UAV-mounted visual detectors have recently and confidently indicated crowd existence on it. In order to calculate prior probabilities for Bayesian fusion, the method also proposes and exploits a simple, but efficient image processing-based algorithm for identifying flat terrain areas (under the assumption that people do not gather on highly curved or inclined terrain), relying on a priori available ground elevation data for the mapped area. Evaluation on both synthetic and real-world multiview video sequences depicting human crowds in outdoor

environments verifies the effectiveness of the proposed method.

Crowd Detection, Drone Vision, Image Processing, Autonomous Drones, Multiview Fusion

1. Introduction

Camera-equipped Unmanned Aerial Vehicles (UAVs, or "drones") are widely employed for a variety of applications, including media production, search and rescue operations, infrastructure inspection, etc. Cognitive autonomy functionalities, such as visual object/target detection and tracking [1] [2] [3] [4], are gaining more and more traction in current commercial UAVs, since they facilitate significantly easier drone deployment and operation [5] [6] [7] [8]. However, safety concerns constitute an obstacle to more widespread adoption of autonomous UAVs, mainly due to the risk they pose to humans in case of malfunction [9].

Drone flight regulations postulate human crowd avoidance: drones are typically not allowed to fly over a crowded area and must maintain a certain safety distance from the crowd. Thus, in autonomous UAVs, the on-board cognitive functionalities should be partially devoted to implementing these policies autonomously. To this end, 2D crowd regions can be detected on video frames using heatmaps, derived through crowd detection approaches that rely on embedded Convolutional Neural Networks (CNNs) [10]. Each heatmap is a grayscale image that spatially corresponds to the RGB video frame it was derived from, but where the luminance value of each pixel represents the probability it depicts human crowd. Similar crowd heatmaps may be exploited, for instance, to facilitate autonomous on-line mission/path replanning, so as to avoid flying over/near crowds. This would lead both to increased conformance to regulations/legislation and to enhanced safety, while also suppressing the need for a human operator being constantly involved.

However, in the common case of camera-equipped UAV fleet members covering different parts of a large-scale outdoors area over time, accurate 3D localization of the on-frame detected human crowd heatmaps is not trivial. Crowd detection errors (per-pixel false positives/negatives) may accumulate over time, observations from different cameras need to be fused, while the possibly dynamic nature of the scene, i.e., moving crowds, further complicates the issue. For instance, the 3D crowd regions derived from the camera stream of each vehicle can naively be accumulated over time and combined using an

OR operator, but this is suboptimal due to 2D visual crowd detection noise and the possibility of dynamically moving human crowds. On top of all these, aerial crowd detection itself (as opposed to crowd density estimation or crowd counting) is a relatively new and unexplored topic, thus relevant literature is rather sparse.

This paper presents a novel centralized method for on-the-fly 3D localization of human crowds, initially detected on video frames (in 2D pixel coordinates) acquired by independent camera-equipped UAVs. A certain 3D area can be viewed by more than one UAV-mounted cameras, at coinciding or different time instances.

The method makes the following assumptions:

- All UAVs communicate with a central Ground Station computer (e.g., via WiFi, 4G/LTE links, etc.). FANET schemes could be used for always maintaining a link between each UAV and the Ground Station, using other fleet members as relays.
- The 3D flight area terrain map is a priori available in a usable form (e.g., Octomap [11]).
- A real-time 2D visual crowd detector is running separately on-board each involved UAV (e.g., the method in [10] executed on a GP-GPU-equipped nVidia Jetson Xavier board), thus on-the-fly transforming each captured video frame into a human crowd heatmap image of identical resolution. Importantly, such heatmaps are susceptible to errors.
- The various heatmaps are timestamped using a common clock, for synchronization purposes.
- Extrinsic/intrinsic parameters are known for all cameras at all times.
- An independent fleet mission/path planning subsystem is in place (e.g., [12]), for coming up with UAV fleet member trajectories and dynamically replanning as needed.

Given this setup and set of assumptions (illustrated in Figure 1), the proposed method's output is a set of consistent semantic annotations of human crowds (in 3D geo-referenced coordinates), being constantly derived per video frame and used for regular updates of the common 3D area map. The proposed method takes care both of suitably fusing the independent observations of

the various fleet members, in order to compensate for the errors present in each one, and of properly localizing them in a common, global 3D space. The goal is to semantically annotate the employed 3D geometric area map used for flight/navigation with the presence of human crowds, so that other algorithms can better adjust mission/path plans on-the-fly for enhanced safety.

The method is composed of three cooperating components: a) the Map Projector (MP), b) the Bayesian Fusion mechanism (BF), and c) the Flat Area Identifier (FAI). Initially, the heatmap corresponding to the currently processed video frame (i.e., a grayscale image in 2D pixel coordinates) is transmitted from each UAV to the central Ground Station. There, crowd regions identified using probability thresholding, separately on each heatmap of each UAV, are back-projected by the Map Projector, via raycasting, onto the common 3D map of the navigation environment. The resulting preliminary projected regions do not take into account previous observations (no fusion over time) and do not intelligently combine the detections coming from different UAVs (no multiview fusion across cameras). Thus, they are subsequently processed by the Bayesian Fusion mechanism, using the stored semantic annotations set of the last time instance. Prior probabilities required for the BF calculations are provided by the Flat Area Identifier component, under the assumption that human crowds are usually concentrated in large and flat terrain areas. The final method output for each time instance/video frame is a set of semantic 3D map annotations that indicate crowd gathering locations, defining no-fly zones in 3D space.

The MP component of the proposed method is an elaborate engineering pipeline relying on pre-existing algorithmic building blocks, while both BF and FAI rely on novel algorithms. MP and BF need to be executed on-line for each video frame, while FAI can be executed once, off-line, before deploying the UAV fleet, since it depends on a priori available, static Digital Elevation Models (DEMs) [13] of the flight environment's terrain. FAI relies on image processing and is not learning-based; thus it does not require enormous training datasets, unlike competing deep learning-based methods. Additionally, a useful by-product of FAI is a set of potential UAV emergency landing sites within the flight environment, under the typical assumption that a drone can only land safely on sufficiently large, non-inclined terrain.

Thus, overall, the contributions of this paper are the following ones:

- An engineering pipeline for on-line semantic 3D map annotation through back-projecting 2D visual detections, via raycasting (MP). Existing

methods with similar goals (e.g., [14, 15]) do not support the multiview setting of UAV fleets and typically rely on RGB-D sensors instead of simple, regular visual cameras [14].

- A novel Bayesian algorithm for on-line multiview fusion of map annotations over time and across cameras (BF). No such algorithm has been proposed before for vision-based multiple-UAV semantic mapping, in contrast to other multiview tasks where Bayesian approaches have been successfully applied (e.g., head pose estimation, pedestrian detection, etc.).

- A novel algorithm for identifying large and flat terrain areas by processing available DEMs (FAI), that provides auxiliary information to the multiview semantic mapping process and identifies potential UAV landing sites, as a by-product. In contrast to competing methods, the proposed algorithm does not require vast training datasets or complicated optimization, but only publicly available DSM and DTM height maps.

The above ingredients are combined into a cohesive system, in order to permit vision-based human crowd on-map localization with acceptably high accuracy, under a UAV fleet setting.

It must be noted that the algorithm in FAI has been previously published in conference form [16], for the narrow purpose of a priori potential UAV landing site detection within the known flight area. Thus, since the goal, context and application domain of [16] is entirely different, the experimental evaluation of FAI performed in this paper significantly deviates from the one in [16]. Here, the main purpose of FAI is to assign prior crowd gathering probabilities, based on assessing ground morphology. These probability estimates are exploited by the BF module, which is presented here for the first time. Therefore, overall, this paper serves as the extended journal version of [16]; the entire system is completed and evaluated as a whole, including all cooperating components that were not presented in [16] (MP and BF).

The remainder of this paper is organized as follows. Firstly, Section 2 presents previous work related to crowd detection and localization from UAV-captured videos. Then, Section 3 describes the proposed method, including its MP, BF and FAI components. Subsequently, the method is evaluated in terms of performance in Section 4, using synthetic multiview video sequences depicting human crowds in outdoor environments as well as real-world drone

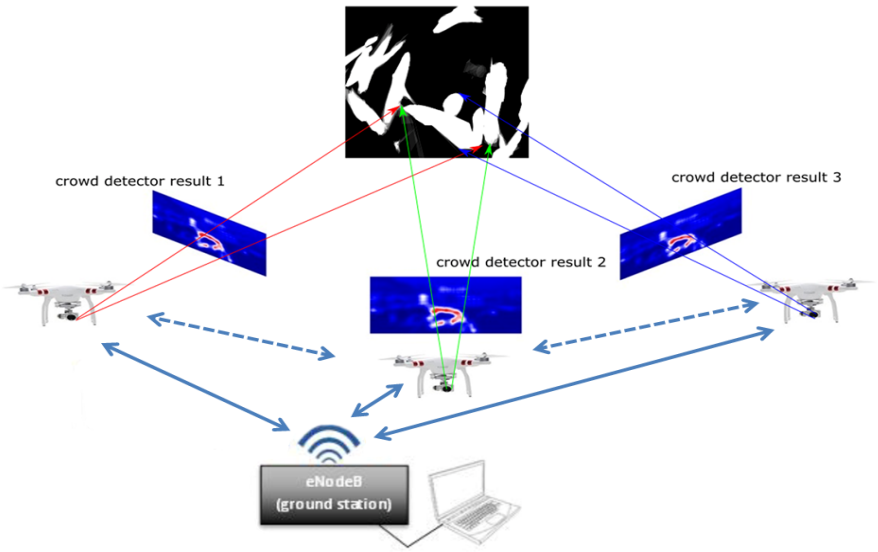


Figure 1: A schematic of the assumed method setup.

data experiments. Finally, conclusions are drawn from the preceding discussion in Section 5.

2. Related Work

Typically, 3D terrain information is represented as 3D occupancy grids. Two different 3D mapping solutions are worth mentioning: a) Octomap [11], and b) Voxblox [17]. The first one provides a probabilistic 3D occupancy map compressed in a hierarchical way for memory efficiency. It consists in a volumetric representation of occupied, free and unknown areas in space, based on octrees. These maps are usually generated by post-processing data from LiDAR (Light Detection And Ranging) surveying. Voxblox, on the other hand, provides a field distance map, i.e., for each voxel, the Euclidean distance to the nearest obstacle is provided.

An alternative source offering terrain information is Digital Elevation Models (DEM) [13], which come in two forms. The first type is Digital Terrain Models (DTM) [18],[19] that includes information regarding the height variations of an area's bare ground, without any man-made structures or vegetation. The second type is Digital Surface Models (DSM) [20],[21]. A

DSM provides a representation of the elevation values for areas of exposed ground, road surfaces, tree crowns, vegetation and buildings. In other words, DSMs include information for both the ground and the man-made structures or vegetation that lie on it. DSMs can be generated by data coming from various sources such as LiDAR (Light Detection And Ranging) surveying. DTMs are usually generated by post-processing DSMs. DSMs and DTMs often come in raster format, i.e., essentially georeferenced images where a pixel's value denotes elevation of the corresponding location.

Several approaches aiming to augment topological maps [22] with semantic information [23] [24] and high-level attributes have been proposed over the past years, allowing aerial robots to handle more expressive concepts or be deployed for more sophisticated tasks. Typically, the goal is to segment the environment into regions that have a coherent semantic meaning. Recently, the robotics community has focused on the presence of semantics in maps [25] [26] [27], to develop autonomous robots capable of understanding the semantic relationships between the objects in the environment [28] [29] [30], besides exploiting occupancy grid maps for navigation.

Although several works utilize deep CNNs for crowd analysis and understanding, e.g. [31, 32, 33], research on crowd detection in drone-captured images is rather a little bit more than an uncharted territory. One reason might be that the aerial point-of-view bears additional challenges (e.g., small person size, occlusions etc.), in comparison to a ground point-of-view. Since the crowd first needs to be detected on-frame, relevant algorithms must be capable of efficiently distinguishing between crowded and non-crowded video frames. An application-tailored deep Convolutional Neural Network is presented in [10], where a pretrained model is finetuned for the task of crowd detection. Moreover, in [34], the authors propose a novel crowd detection method for drone safe landing, based on an extremely lightweight and fast Fully Convolutional Neural Network.

A different approach in [35] performs sampling-based multiview crowd detection for accurate localization of people in 3D space, based on single or multiview images. In addition, a probabilistic method for accurate 3D crowded scene localization and cross-view tracking is presented in [36], where binary constraints are exploited to localize truncated human boxes in a 3D map and identify moving subjects, exploiting a Bayesian approach for handling failure cases (e.g., occlusions, false detections). Finally, [37] presents an algorithm for detecting coherent crowd groups utilizing a Multiview-based Parameter-Free framework (MPF). This framework consists of a novel struc-

tural context descriptor for individual crowd characterization, a multiview clustering approach for grouping these descriptors, as well as a module that automatically determines the size (in terms of people contained therein) of each crowd area.

Despite the prominence of Bayesian methods in general data fusion, literature on Bayesian multiview visual information fusion is rather limited. A Bayesian filtering approach for multiview head pose estimation is presented in [38]. The method fuses neural network outputs from multiple camera views. Bayesian multiview filtering was also used for robust multiple camera pedestrian detection and fake pedestrian detection removal in [39]. Finally, Bayesian multiview filtering for body orientation estimation based on silhouette information in a smart room environment is presented in [40]. Overall, however, none of these methods applies Bayesian filtering to a UAV-related setting or to achieve on-the-fly, on-line multiview semantic area mapping, as proposed in this paper.

Regarding the identification of large and flat areas on a known terrain, the most relevant pre-existing research concerns algorithms for UAV landing site detection. In our case, identification of potential landing sites is simply a useful by-product, but the methodology is similar. In [41], a method is presented for detecting fixed-wing UAV landing sites using the average height and height variance inside quadtree-based DEM partitions. Partitions whose height variance is below a limit are selected as landing sites and merged with neighboring ones with similar average heights. In [42], suitable landing areas are determined on topographical maps for emergency landing of UAVs by utilizing surface fitting on coarse elevation models using Least Squares Error and slope calculation. Furthermore, in [43], the authors create a system for efficient and reliable assessing the safety of landing zones covered in low vegetation by combining a volumetric occupancy map with a 3D Convolutional Neural Network (CNN). In [44] the authors propose a system for landing zone selection based on a relatively simple geometric analysis of terrain roughness and slope. Finally, [45] proposes a scheme for the selection or validation of landing zones for unmanned helicopters with terrain assessment incorporating factors such as terrain/vehicle interaction, wind direction and mission constraints.

3. Proposed Method

The proposed method assumes that all UAVs communicate with a central Ground Station computer (e.g., via WiFi, 4G/LTE links, etc.) and that the 3D flight area terrain map is a priori available. The method's outputs are semantic annotations of human crowds (in 3D geo-referenced coordinates), being constantly derived per video frame and used for regular updates of the common 3D area map. The proposed method assumes that a real-time 2D visual crowd detector (e.g., [10]) is running separately on-board each involved UAV, thus on-the-fly transforming each captured video frame into a human crowd heatmap image of identical resolution. Additionally, the various heatmaps are assumed to be timestamped using a common clock, for synchronization purposes, while extrinsic/intrinsic parameters are assumed to be known for all cameras at all times. Finally, the method requires a DEM of the flight/navigation area to be available, while the 3D geometric area map to be semantically annotated is assumed to be an Octomap (although this is not strictly necessary).

The method is composed of three cooperating components: a) the Map Projector (MP), b) the Bayesian Fusion mechanism (BF), and c) the Flat Area Identifier (FAI). These components are described below in detail and illustrated in Figure 2.

3.1. Map Projector

Let us assume there are N UAVs equipped with a camera and a 2D visual crowd detector system. The crowd detector converts on-the-fly each video frame into a heatmap: a single-channel image of identical resolution, where each pixel luminance value denotes the probability that it indeed depicts human crowd. Thus, heatmap pixel values lie within the real range $[0, 1]$, where a value of 1.0/0.0 implies absolute detector confidence that human crowd is/is not depicted there, respectively, while a pixel value of 0.5 implies maximum detector uncertainty.

At each time instance, the heatmap corresponding to the currently processed video frame (i.e., a grayscale image in 2D pixel coordinates) is transmitted from each UAV to the Map Projector (MP) method component, residing in the central Ground Station. There, each pixel may be back-projected onto the stored 3D terrain using the known extrinsic and intrinsic camera parameters. Such an approach is presented in Section 3.1.1, detailing a ray-casting method to annotate the 3D occupancy grid of an Octomap with

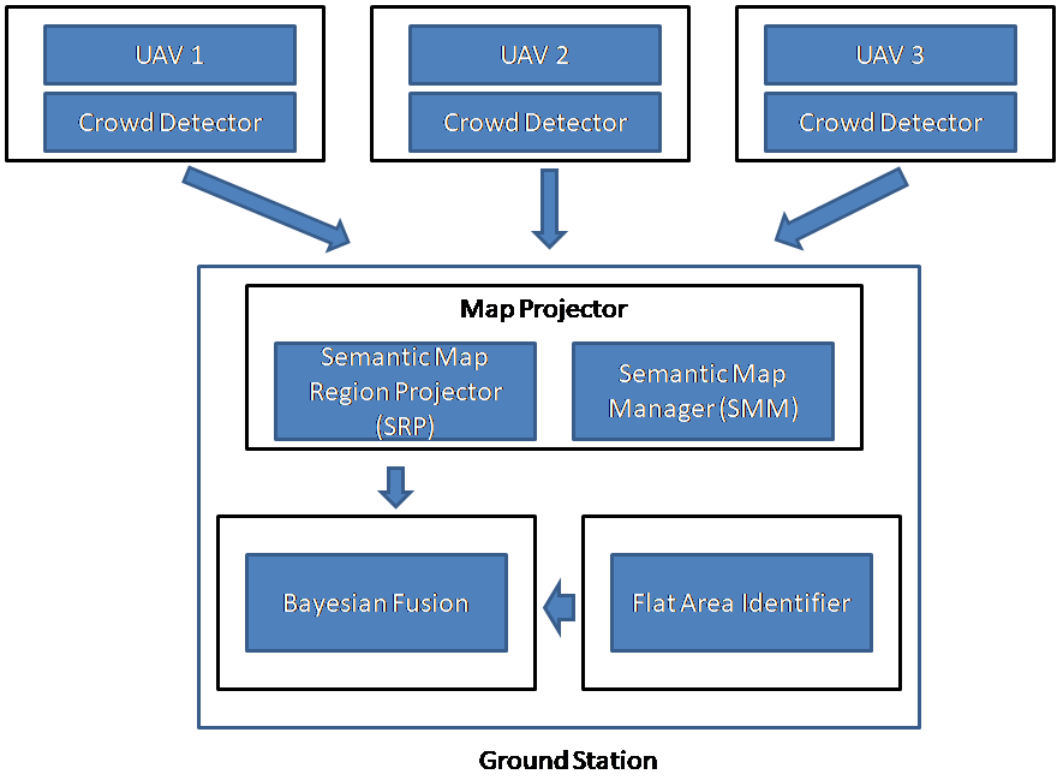


Figure 2: Block diagram of the proposed algorithm.

human crowd information.

3.1.1. Semantic Map Region Projector (SRP)

The Semantic Map Region Projector (SRP) receives and temporally synchronizes each of the N heatmaps. Subsequently, by thresholding them in order to retain only image locations with high probabilities of crowd existence, they are converted into binary images where groups of adjacent pixels with value 1 (white) represent 2D regions occupied by crowd. Next a contour-following algorithm is applied in order to find the contours of this image, resulting in a new binary image indicating the boundaries (white pixels) of the aforementioned crowd regions 2D polygons. If needed, the polylines are simplified maintaining their shape according to the Ramer-Douglas-Peucker algorithm [46], which takes a curve composed of line segments and finds a similar one with fewer points.

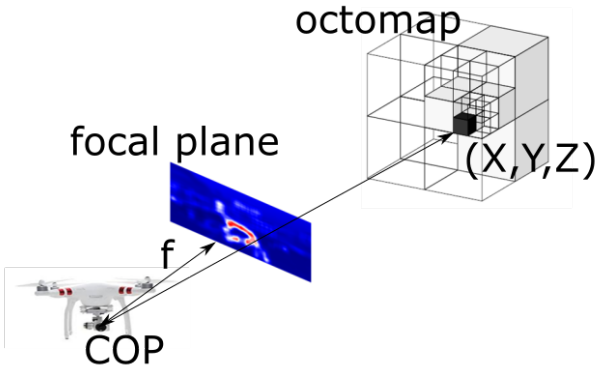


Figure 3: Ray casting procedure for crowd heatmap contours projection onto the 3D occupancy grid of Octomap as semantic annotations.

Finally, by traversing the points (pixels) of the regions' boundaries in a counter-clockwise manner, ray casting is employed for back-projection onto the 3D volumetric map handled by Octomap [47], [48]. The contour image lies on the focal plane of the UAV camera, whose parameters are fully known. Thus, one may cast a ray from each of the boundary contour points towards the voxels of the Octomap. This results in finding the occupied voxel hit by each ray, leading to the evaluation of the X, Y, Z terrain coordinates where

each of the contours' points is projected, as the Octomap is coordinate-referenced. Since the 2D boundary contour points are traversed sequentially, so are the points of the 3D boundary contour (polyline).

3.1.2. Semantic Map Manager (SMM)

After the SRP, the second stage of the MP's operation consists of the Semantic Map Manager (SMM), whose functionality can be summarized as follows: firstly, the created polylines which delineate crowd gathering locations on the 3D map are merged with previous ones, as the UAV moves and its camera "sees" new areas of the terrain. This is done using the union operator, as a preliminary stage of crowd information merging. Subsequently, the constantly updated geometrically localized semantic annotations are stored in an internal data layer.

Overall, the output of the MP for the current time instance and corresponding set of synchronized video frames is a preliminary set of human crowd map annotations which do not take into account previous observations (no fusion over time) and do not intelligently combine the detections coming from different UAVs (no multiview fusion across cameras).

3.2. Bayesian Fusion

After the MP processing step, the Bayesian Fusion (BF) procedure is executed. During this computational step, on-the-fly fusion of projected MP crowd annotations from several UAVs is conducted, before advancing to the next time instance. The main aim of this module is to increase crowd localization accuracy by taking into account the dynamic nature of the crowd detection outputs and by compensating for the 2D crowd detector's noise. The novel Bayesian multi-view fusion algorithm is explained below.

Let us define the following Bernoulli-distributed (binary event space) random variables (RVs):

A: random variable denoting crowd presence in a voxel of the Octomap.

B_i: random variable denoting crowd detection in the corresponding pixel on the heatmap of the i-th UAV.

Due to the combination of N hypotheses on the same underlying truth, i.e., the estimation of a voxel probability containing crowd based on N different probability models, the posterior probability of a voxel actually containing crowd can be derived according to Bayes theorem and Linear Opinion

Pool [49]:

$$P\left(A=1|B_1=b_1,...,B_N=b_N\right)= \sum_{i=1}^{N} w_i \frac{P(B_i=b_i|A=1)P(A=1)}{P(B_i=b_i|A=1)P(A=1)+P(B_i=b_i|A=0)P(A=0)}, \quad (1)$$

where $b_i \in \{0,1\}$ (non-crowd/crowd) is the detector's binary observation from the i-th UAV, derived by thresholding its raw probability output o_i.

In the above:

- $P(B_i=1|A=1)$ is the detector's True Positive Rate (TPR)
- $P(B_i=1|A=0)$ is the detector's False Positive Rate (FPR)
- $P(B_i=0|A=1)$ is the detector's False Negative Rate (FNR)
- $P(B_i=0|A=0)$ is the detector's True Negative Rate (TNR). All these four probabilities can be evaluated experimentally for the specific crowd detector being viewed.
- $P(A=1)$ is the a-priori probability of a certain location (voxel) to be occupied by crowd. Assuming that the human crowds would only gather in large, relatively flat/non-inclined terrain areas (see Section 3.3), this probability can be considered equal to the percentage of the area surface that satisfies these criteria.

The choice of using Linear Opinion Pool against other opinion pools, such as Log-Linear, is justified from the fact that the former provides a desirable property, which is invariance to event combination [50]. Moreover,the fusion aims to combine N candidate crowd probabilities even if a subset of them is zero. However, the Log-Linear Opinion Pool is unimodal [51] and if even a single crowd probability is zero, then the combined probability will be zero too. This is not the case with the Linear Opinion Pool, where a null probability would be averaged with all other ones in a weighted manner.

Thus, weights w_i encoding the degree to which we take each detector's decision into account regarding the currently processed voxel are denoted in Eq. (2). When the detector outputs (either positive or negative ones) are confident, this weight should be higher. Given the above, w_i can be determined as the absolute difference between the original detector's output percentage m_i from 0.5:

$$w_i = \frac{|m_i - 0.5|}{\frac{1}{2}N}, \tag{2}$$

where, for voxels on which crowd has been projected, m_i is their raw probability output (derived by averaging the o_i probability values of the corresponding heatmap pixels), while for voxels where no crowd has been projected it holds that $m_i = 0$. The denominator in Eq. (2) is required so as to keep all probabilities in Eq. (1) valid (i.e., lying within the interval $[0, 1]$).

A crowd annotation forgetting policy is also exploited in the current method. Gradual forgetting of old detections for each voxel can be introduced by Eq. (1). More precisely, proper weighting over time of each UAV's contributions (e.g., by using a very slow exponential decay), is added, if no newer observation are made regarding that specific voxel. Any new observations can either override the old ones (a new posterior probability is computed from scratch, referred to as Fusion Scenario 1), or be merged with them by properly extending Eq. (1) with additional terms in the sum (Fusion Scenario 2). Both temporal fusion policies compute an aggregate/fused probability $\tilde{P}_t$ of a voxel actually containing crowd at current time instance t, using the following probability blending formula:

$$\begin{aligned} \tilde{P}_t = (1 - \alpha\gamma_t)\beta_{\Delta t}\tilde{P}_{t'} + \\ +\alpha\gamma_t P\left(A = 1 | B_1 = b_1, ..., B_N = b_N\right)_t, \end{aligned} \tag{3}$$

where t' is the last time instance the voxel was visible through any of the UAVs/cameras, $\Delta t = t - t'$, λ is a temporal decay rate hyperparameter, $\beta_{\Delta t} = e^{-\lambda \Delta t}$ and $\tilde{P}_{t'}$ is the last stored aggregate probability that the voxel in question contains crowd, as computed in previous time instance t'. Also, $\alpha \in (0, 1]$ is a hyperparameter regulating the degree to which the current/newest observations override older ones. $\gamma_t \in \{0, 1\}$ is not a parameter, but a binary value denoting whether the voxel being currently processed is visible by at least one UAV at the present time instance t. Thus, $[1 - \alpha\gamma_t]$ evaluates to $1 - \alpha$ when at least one new observation is currently available ($\gamma_t = 1$) and to 1 otherwise ($\gamma_t = 0$).

The current fused probability of crowd existence is computed according to Eq. (1), by employing the detections of current time instance t (if the corresponding area is visible by at least one UAV). Overall, setting α to 1 results in Fusion Scenario 1, while setting it to a real value in the interval

$(0, 1)$ results in Fusion Scenario 2.

Using the above setup, the presence of human crowd in an area which is denoted as a no-fly zone, is considered to be certain if and only if all or most of N UAV-mounted visual detectors concurrently detect it at the present moment with high probability. In case only a subset of the UAVs detect crowds and/or a significant time interval has lapsed since last crowd detection in that area, the output aggregate/fused probability falls dramatically.

The proposed BF module/method innovates by applying a simple Bayesian formula and a Linear Opinion Pool, in order to promptly and efficiently fuse human crowd probability heatmaps on-the-fly, under a multiple-UAV setting. To the best of our knowledge, no such Bayesian on-line method has been previously presented.

3.3. Flat Area Identifier

Unlike MP and BF, which need to be executed on-line for each video frame, Flat Area Identifier (FAI) can be executed once, off-line, before deploying the UAV fleet, since it relies on a priori available, static Digital Elevation Models (DEMs) [13] of the flight environment's terrain. It employs a novel algorithm which computes prior probabilities required for the BF calculations in Eq. (1), but also exports potential UAV landing sites as a useful by-product. This dual use results from the main goal of FAI, i.e., to identify (sufficiently) flat and large areas within the flight area. Such regions can be used as normal or emergency UAV landing points, while also serving as potential human crowd gathering spots, due to friendly ground morphology.

The employed method (previously published in conference form [16], where it was evaluated only for potential UAV landing site detection) utilizes the information contained in pre-obtained Digital Terrain Model (DTM) and Digital Surface Model (DSM) files. Based on this, it detects the vegetation, buildings and generally the objects upon the bare ground, by evaluating the height difference between the DTM and DSM models.

The algorithm's input consists of the two flight area DEMs, namely the DSM and the DTM, in raster format, i.e., as a regular grid of elevation values. The DTM (Figure 4-a) only depicts the terrain and no man-made structures or vegetation, whereas the DSM (Figure 4-b) depicts the terrain along with buildings and vegetation. DSM files often contain pixels with no value (no elevation information), as a result of sensor inefficiencies during DSM acquisition. As DTM is constructed by post-processing the DSM, these pixels are usually assigned values through interpolation. Here, DSM pixels

with no values are assigned elevation values from the corresponding pixels of the DTM file. Flat areas are discovered by evaluating the local terrain slope through estimating the image gradient on the DEM file and thresholding the gradient magnitude image, so as to retain areas having small local 3D gradient. Connected component analysis is applied on the resulting binary image, so as to identify and retain regions whose area is above a preset threshold. The final map is constructed by combining the results of building and vegetation detection with the results of the previous step, delineating regions that are sufficiently large and flat.

Overall, the FAI algorithm can be summarized in five steps:

1) *Detection of man-made structures and vegetation*: by subtracting the DTM from the DSM and applying a threshold to the outcome, a binary image is obtained (Figure 4-f) which marks pixels depicting man-made structures and vegetation whose height is above a selected (small) threshold.

2) *Terrain slope determination* (Figure 4-g): the local slope of the depicted areas in the DSM is calculated. According to Geographic Information Systems (GIS) theoritical definition [52], the slope is the maximum rate of change in value (elevation) from a pixel (cell) to its neighbors. The lower the slope value, the flatter the terrain. As far as the slope calculation is concerned, the rates of change of the surface elevation in the horizontal ($\frac{dz}{dx}$) and vertical ($\frac{dz}{dy}$) directions from the central cell determine the slope. Slope, in degrees, is calculated as [52]:

$$slope_{degrees} = \frac{180}{\pi} \arctan \sqrt{\left(\left[\frac{dz}{dx} \right]^2 + \left[\frac{dz}{dy} \right]^2 \right)} \tag{4}$$

The values of the center cell and its eight neighbors determine the horizontal and vertical rates of elevation change. For a neighbourhood such as the one depicted in Figure 5, the rates of change along the x and y direction for cell "e" can be calculated as:

$$\frac{dz}{dx} = \frac{(c + 2f + i) - (a + 2d + g)}{8 * x_{cellsize}} \tag{5}$$

$$\frac{dz}{dy} = \frac{(g + 2h + i) - (a + 2b + c)}{8 * y_{cellsize}} \tag{6}$$

Essentially, the rates of change along the x and y direction are the hori-

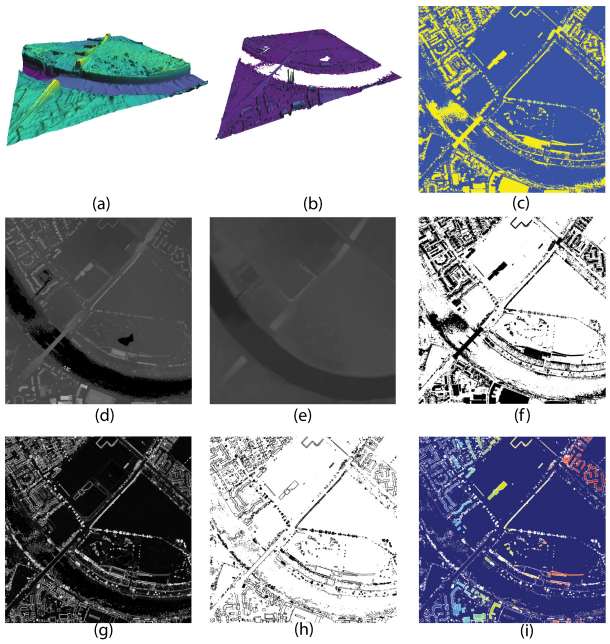


Figure 4: Example of processing DEMs: (a) 3D view of DTM, (b) 3D view of DSM, (c) final map (meaning of colors is explained in the text) (d) DSM, (e) DTM , (f) binary image of buildings/vegetation, in black, (g) Terrain slope determination via Sobel operator, (h) binary image of low slope areas, in white, (i) connected components analysis result.

a	b	c
d	e	f
g	h	i

Figure 5: 8-neighborhood of a DSM.

zontal and vertical derivative approximations, generated by the well known Sobel operator [53] scaled by a factor of $-8 * cellsize$.

3) *Sobel operator gradient image thresholding* (Figure 4-h): after extracting the elevation gradient magnitude image, it is thresholded in order to classify the DSM pixels in "flat" or "non-flat" areas based on the local slope. Obviously, near-flat areas are retained. The terrain slope threshold may be adjusted as a hyperparameter, according to desired tolerance to inclination.

4) *Binary image connected components evaluation* (Figure 4-i): connected components analysis is applied on the binary image resulting from the pre-

vious step. Connected components with sufficiently large number of pixels, i.e., of sufficient area, are retained.

5) *Creation of the final map* (Figure 4-c): in order to create the final map, any parts that overlap with buildings and vegetation (found in Step 1) are removed from the large low slope areas (found in the previous Step). The final map consists of three categories of pixels:

- Blue pixels: this category of pixels corresponds to regions in the DSM map with small terrain slope and large enough area.
- Light blue pixels: this category of pixels corresponds to regions in the DSM map with large terrain slope or very few pixels (small area).
- Yellow pixels: these pixels correspond to buildings and vegetation.

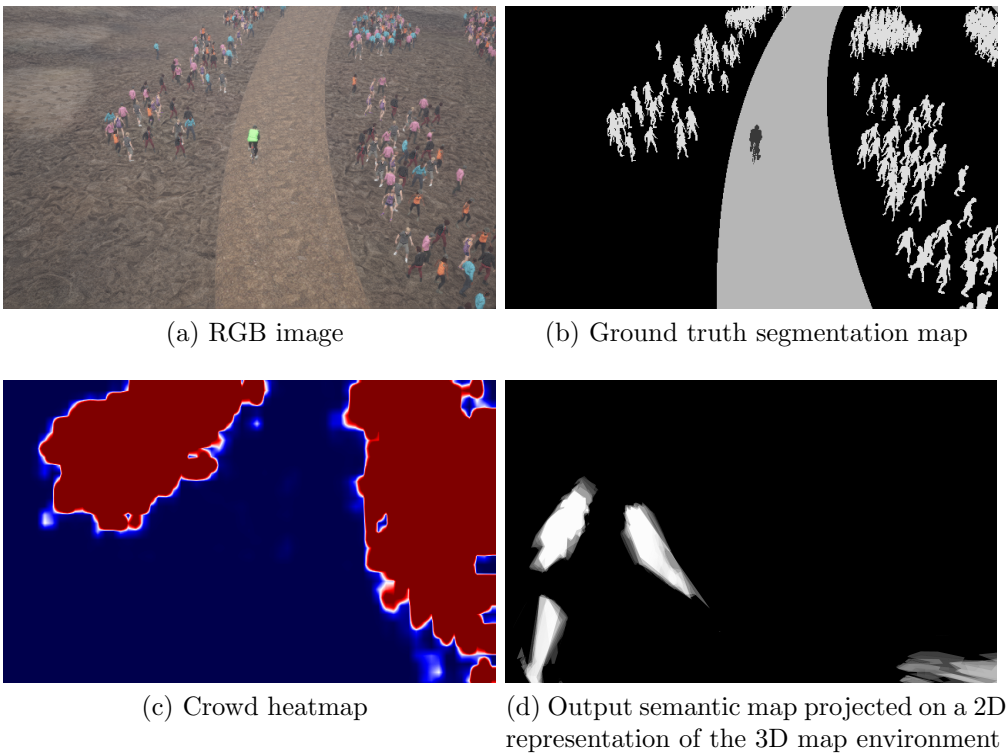
4. Experimental Evaluation

The proposed method was fully implemented in C++ using the well-known Robotic Operating System (ROS) [54] middleware. ROS allows easy InterProcess Communication under the publisher-subscriber model, in a distributed setting. The employed ROS messages were: a) a *drone telemetry* message, containing the UAV GPS coordinates, b) a *gimbal status* message, containing the gimbal pitch, roll and yaw, as well as c) a *camera status* message, containing the camera focal length and sensor width/height. All these messages were synchronized with the video frames according to timestamps. The pre-trained 2D visual crowd detection CNN from [10] was also employed, in the form of a separate ROS process.

Two experimental setups were used to assess the proposed multiview human crowd localization approach: a simulated one and one based on real-world UAV flights. They are both described below, before the presentation and discussion of evaluation results.

4.1. Synthetic drone data experiments

In the simulation evaluation setup, synthetic multiview video sequences depicting human crowds in outdoor environments and captured by $N = 3$ simulated UAVs were constructed using the Microsoft AirSim simulator, built on top of the real-time 3D graphics engine Unreal Engine 4 (UE4). All video frames were stored along with the corresponding ROS messages. A mountainous, large-scale UE4 terrain model was selected, covering an area



(a) RGB image

(b) Ground truth segmentation map

(c) Crowd heatmap

(d) Output semantic map projected on a 2D representation of the 3D map environment

Figure 6: Samples of the multiview human crowd aerial synthetic image dataset and the proposed method output.

of approximately 1270×1017 meters within the virtual world, and multiple crowds were placed on the sides of a road serving as the path of a bicycle race. The crowds were designed to move slowly along the road while keeping their cohesion. Using AirSim, three UAVs were deployed to follow three cyclists (one drone per cyclist), being quite far apart for one another. In this evaluation setup, coordinated multiple-UAV cinematography shooting trajectories were planned according to [8]; in a different scenario, any UAV fleet mission/path planning subsystem would also be admissible, as long as the same 3D area is viewed by more than one UAV-mounted cameras, at coinciding or different time instances.

Overall, three annotated video sequences were created (one per drone), each one containing more than $K = 1056$ frames. The annotations included the center of each crowd on the frame and a per-pixel ground truth segmenta-

tion map of the frame, containing the following five classes: crowd, ground, sky, road, cyclist. An example video frame, along with its ground-truth segmentation map, the respective human crowd heatmap and the output semantic area map of the proposed MP+BF+FAI pipeline, are shown in Figure 6. For visualization purposes, the semantic 3D area map is projected onto the 2D (top-down) representation of the 3D Octomap.

The ground-truth for human crowd presence on 3D terrain was extracted from the available ground-truth segmentation maps (one per video frame, with a spatial resolution of 640×360 pixels). Crowd-class pixels in segmentation maps were processed into connected components, through merging individual silhouettes and filling in gaps. These connected components were then projected onto the 3D terrain. The mean TPR (True Positive Rate), FPR (False Positive Rate), FNR (False Negative Rate) and TNR (True Negative Rate) in Eq. (1) were calculated by comparing the thresholded crowd heatmaps produced by [10] and the binary ground-truth, on the validation set of the dataset where the model was trained.

4.2. Real-world drone data experiments

An additional set of experiments was performed, using data from real-world UAV flights, in order to better validate the proposed method. To this end, $N = 2$ 2-minute drone video sequences (of spatial resolution equal to 1920×1080 pixels) were simultaneously captured, containing a simply structured human crowd (group of six persons) located on a flat terrain and moving slowly, while preserving its cohesion, as depicted in Figure 7. All necessary camera parameters and captured video frames were stored during flight, in order to facilitate not only method execution, but also reproducibility of the experiments. An Octomap file was additionally constructed, representing the flat terrain upon which the filmed crowd was placed. Finally, a person within the target crowd carried a GPS device in order to record its ground-truth real-world position. This was selected as a simpler but feasible alternative to obtaining accurate ground-truth locations for the entire crowd region in the real world, which was very difficult to achieve.

4.3. Performance Evaluation

Performance of the proposed multiview crowd heatmap fusion is detailed below, for both the synthetic/simulated and the real-world UAV flight setups. In order to validate the importance of the novel BF and FAI components of the proposed method, a "no-fusion" variant was also evaluated which

(a) Video frame captured from drone No 1. (b) Video frame captured from drone No 2.

(c) Qualitative view of the cumulative visualization output over time, depicting identified crowd-annotated areas ($\alpha = 1$ and $\lambda = 0$). The predicted/ground-truth crowd location is shown as the yellow polyline/green location pointer, respectively (in real scale).

Figure 7: Real-world experimental setup, filming a simply structured human crowd.

accumulates the 3D annotations derived from all drones over time and simply combines them using an OR operator. Thus, the "no-fusion" variant is more of an elaborate engineering pipeline [55].

Performance of the proposed multiview crowd heatmap fusion in the synthetic/simulated data evaluation setup was measured by Intersection-over-Union (IoU) on the projected crowd detections. The mean IoU over all video frames was computed, so as to observe the performed annotation accuracy

of the crowd regions. It is given by:

$$IoU_{mean} = \frac{1}{K} \sum_{i=1}^{K} \frac{Overlap_i}{Union_i}, \tag{7}$$

where K is the total number of video frames, while $Overlap_i$, $Union_i$ are the overlap and the union area, respectively, between the ground-truth information and the projection of crowd prediction regions onto the terrain.

Given that the field-of-view for each camera at any time instance covers only a part of the map/flight area, at each time there may exist map regions not yet seen by any drone. Therefore, the current crowd annotations derived by the proposed method cannot be directly compared against the complete ground-truth map. Thus, in this paper, the IoU for each video frame was computed in the following manner: only the union of the projections of the camera fields-of-view on the map from all drones at all time instances from mission start up to that time instance is taken into account; not the complete map.

Semantic annotation performance on real-world drone data experiments can only be measured offline, after the experiment is conducted by following the specified protocol. Two evaluation modes were employed:

- First, an objective evaluation was conducted, using ground-truth information concerning the geolocation of the depicted crowd. The employed evaluation metric was a boolean indicator, estimating whether the ground-truth position of the crowd centre, as recorded by the GPS device held by a member of the crowd, falls within the predicted area annotations.

- Second, a subjective evaluation was conducted on a sample of the material using $M = 5$ subjects, exploiting recorded ground-truth information. The subjects were given separately visualizations of the semantic map polygons obtained in two ways: a) the no-fusion approach (MP-only) and b) the complete proposed method (MP+BF+FAI). Subjects were independently shown the two method results (the semantically annotated map) and the original video files. Then, they were asked to rate the results in terms of their perceived quality, along a scale graded the following way: poor (0% - 40%), fair (40% - 60%), good (60% - 75%), very good (75% - 90%) and excellent (90% - 100%).

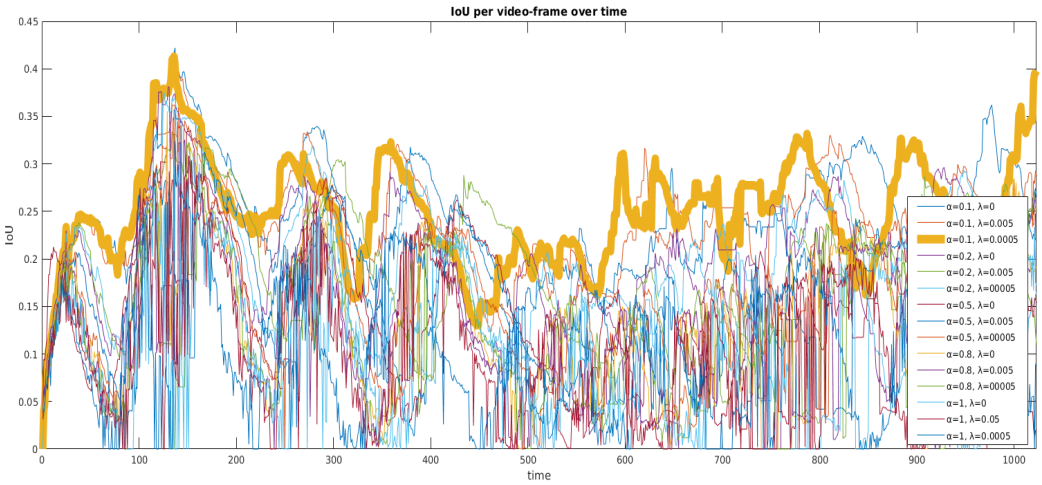


Figure 8: IoU per video frame against time. The best-performing variant of the proposed method is highlighted in bold.

4.4. Evaluation results

Results of the experimental evaluation process are detailed below, both for the synthetic and for the real-world setup.

4.4.1. Evaluation on synthetic data from simulated UAV flights

Evaluation results are illustrated in Fig. 8 and in Table 1, which depicts the mean IoU and the standard deviation of the IoU values, measured according to the process detailed in Subsection 4.3. Evidently, the proposed method outperforms the naive no-fusion approach for the optimal values of parameters α and λ. The maximum mean IoU value is obtained in the case of probability blending with $\alpha = 0.1$, which essentially means that at each time instance the current map annotations evaluated using Eq. (1) contribute by 10%, whereas the last stored map annotations contribute by the remaining 90%. Increasing α beyond this value (including $\alpha = 1$, which is equivalent to Fusion Scenario 1) leads to algorithm performance deterioration. Regarding λ, one can notice that this parameter has a much less pronounced effect and that the best results are typically obtained by using a very small value, i.e., $\lambda = 0.0005$.

The plot in Fig. 8, which shows per-frame IoU over time, verifies the above results. Each per-frame IoU value is computed only within the union of all drone cameras' fields-of-view, after their projection on the area map,

Table 1: Synthetic experimental evaluation results (mean IoU).

Method	**mIoU** ($\pm std$)
no-fusion [55]	0.1271 ($\pm$0.0872)
multiview $a = 0.1, \lambda = 0$	0.2403 ($\pm$0.0612)
multiview $a = 0.1, \lambda = 0.05$	0.2394 ($\pm$0.0578)
multiview $a = 0.1, \lambda = 0.0005$	**0.2488** ($\pm$0.0562)
multiview $a = 0.2, \lambda = 0$	0.1992 ($\pm$0.0723)
multiview $a = 0.2, \lambda = 0.05$	0.1825 ($\pm$0.0719)
multiview $a = 0.2, \lambda = 0.0005$	0.1957 ($\pm$0.691)
multiview $a = 0.5, \lambda = 0$	0.1434 ($\pm$0.0769)
multiview $a = 0.5, \lambda = 0.05$	0.1532 ($\pm$0.0771)
multiview $a = 0.5, \lambda = 0.0005$	0.1582 ($\pm$0.0790)
multiview $a = 0.8, \lambda = 0$	0.1336 ($\pm$0.0742)
multiview $a = 0.8, \lambda = 0.05$	0.1286 ($\pm$0.0759)
multiview $a = 0.8, \lambda = 0.0005$	0.1281 ($\pm$0.0764)
multiview $a = 1, \lambda = 0$	0.1062 ($\pm$0.0767)
multiview $a = 1, \lambda = 0.05$	0.1069 ($\pm$0.0748)
multiview $a = 1, \lambda = 0.0005$	0.1053 ($\pm$0.0745)

aggregated from mission start up to current video frame. Thus, map areas which have never been seen by any drone until the current time instance, are excluded from IoU computation.

Beyond mean IoU, we also employed the mean F-measure metric (averaged over time) for additional evaluation of the proposed method. Precision and recall were calculated in a rectangular grid of the 2D (top-down) representation of the 3D terrain map, for several different grid resolutions (Tables 2,3,4). If any map voxel falling within each patch of this grid is annotated as "crowd" by the proposed method, then the entire patch is assumed to contain human crowds. In most applications, such a setup would most likely be more suitable, since an entire region containing humans should be avoided by the mission/path planner for safety reasons. Unsurprisingly, the results indicate best performance (in terms of mean F-measure) for the highest grid resolution/smallest patches. Once more, the BF and FAI components prove to be crucial for method performance.

Table 2: Synthetic experimental evaluation results (mean F-measure), grid resolution 25×25 patches.

Method (Resolution 25×25 patches)	**m F-measure** ($\pm std$)
no-fusion [55]	0.2152 ($\pm$0.1064)
multiview $a = 0.1, \lambda = 0$	0.3769 ($\pm$0.0826)
multiview $a = 0.1, \lambda = 0.05$	0.3748 ($\pm$0.0798)
multiview $a = 0.1, \lambda = 0.0005$	**0.3884** ($\pm$0.0803)
multiview $a = 0.2, \lambda = 0$	0.3166 ($\pm$0.1072)
multiview $a = 0.2, \lambda = 0.05$	0.2891 ($\pm$0.1077)
multiview $a = 0.2, \lambda = 0.0005$	0.3115 ($\pm$0.1013)
multiview $a = 0.5, \lambda = 0$	0.2284 ($\pm$0.1170)
multiview $a = 0.5, \lambda = 0.05$	0.2434 ($\pm$0.1166)
multiview $a = 0.5, \lambda = 0.0005$	0.2544($\pm$0.1218)
multiview $a = 0.8, \lambda = 0$	0.2236 ($\pm$0.1120)
multiview $a = 0.8, \lambda = 0.05$	0.2168 ($\pm$0.1157)
multiview $a = 0.8, \lambda = 0.0005$	0.2161 ($\pm$0.1151)
multiview $a = 1, \lambda = 0$	0.1943 ($\pm$0.1117)
multiview $a = 1, \lambda = 0.05$	0.1939 ($\pm$0.1109)
multiview $a = 1, \lambda = 0.0005$	0.1965 ($\pm$0.1108)

4.4.2. Evaluation from real-world UAV flights

Objective evaluation results for the real-world UAV flight setup, measured, according to the relevant process detailed in Subsection 4.3, are depicted in Figure 7-c. It is a Google Maps visualization, where the ground-truth region is depicted as a green location marker and the predicted crowd area as a small yellow polyline. As it can be seen, the predicted crowd polyline successfully contains the crowd ground truth pin-point.

Subjective evaluation was conducted according to the relevant protocol detailed in Subsection 4.3. A sample of the content evaluated by the subjects is shown in Fig. 7. One can notice that the identified human crowd annotation area is differently shaped in the output KML file visualization (although correctly localized), in comparison to the intuitive shape of the human crowd depicted in the video sequence. This difference results from the deformation introduced by the projection viewing angle, but is gradually minimized over time as the identified region is progressively getting well-shaped. Overall, the mean perceived quality was 86.6% for the output semantic annotation

Table 3: Synthetic experimental evaluation results (mean F-measure), grid resolution 40×40 patches.

Method (Resolution 40×40 patches)	**m F-measure** ($\pm std$)
no-fusion [55]	0.2334 ($\pm$0.1117)
multiview $a = 0.1, \lambda = 0$	0.3803 ($\pm$0.0808)
multiview $a = 0.1, \lambda = 0.05$	0.3796 ($\pm$0.0781)
multiview $a = 0.1, \lambda = 0.0005$	**0.3905** ($\pm$0.0757)
multiview $a = 0.2, \lambda = 0$	0.3227 ($\pm$0.1038)
multiview $a = 0.2, \lambda = 0.05$	0.2979 ($\pm$0.1057)
multiview $a = 0.2, \lambda = 0.0005$	0.3570 ($\pm$0.0886)
multiview $a = 0.5, \lambda = 0$	0.2391 ($\pm$0.1180)
multiview $a = 0.5, \lambda = 0.05$	0.2534 ($\pm$0.1176)
multiview $a = 0.5, \lambda = 0.0005$	0.2621 ($\pm$0.1197)
multiview $a = 0.8, \lambda = 0$	0.2391 ($\pm$0.1180)
multiview $a = 0.8, \lambda = 0.05$	0.2232 ($\pm$0.1166)
multiview $a = 0.8, \lambda = 0.0005$	0.2220 ($\pm$0.1166)
multiview $a = 1, \lambda = 0$	0.2002($\pm$0.1150)
multiview $a = 1, \lambda = 0.05$	0.1984 ($\pm$0.1128)
multiview $a = 1, \lambda = 0.0005$	0.1975 ($\pm$0.1127)

derived by the complete proposed method (MP+BF+FAI), while it was only 84% for the result of the naive no-fusion method (higher is better).

Subjective evaluation in the real-world setting confirmed the importance of BF and FAI modules, that was already evident from the synthetic evaluation results. The complete proposed method surpassed once more in performance the naive no-fusion alternative.

4.5. Runtime Requirements and Computational Complexity

Regarding runtime requirements, the on-line components of the proposed method (MP and BF) jointly required 1.32985 secs per video frame on average for $N = 3$, while the off-line FAI component required 1.4563 secs once, before mission start. Evaluation was performed on a typical desktop computer with an Intel Core i7-6700HQ CPU @ 2.60Ghz and 16GB RAM. Thus, given a powerful server computer, the method may run stably at a map update rate of approximately 1Hz for $N = 3$. This frequency is enough for typical multicopter UAV flight speeds [5, 6, 1]. Computational complexity is

Table 4: Synthetic experimental evaluation results (mean F-measure), grid resolution 100×100 patches.

Method (Resolution 100×100 patches)	**m F-measure** ($\pm std$)
no-fusion [55]	0.2365 ($\pm$0.1169)
multiview $a = 0.1, \lambda = 0$	0.3841 ($\pm$0.0793)
multiview $a = 0.1, \lambda = 0.05$	0.3833 ($\pm$0.0777)
multiview $a = 0.1, \lambda = 0.0005$	**0.3941** ($\pm$0.0737)
multiview $a = 0.2, \lambda = 0$	0.3262 ($\pm$0.1017)
multiview $a = 0.2, \lambda = 0.05$	0.3017 ($\pm$0.1046)
multiview $a = 0.2, \lambda = 0.0005$	0.3613 ($\pm$0.0863)
multiview $a = 0.5, \lambda = 0$	0.2424 ($\pm$0.1181)
multiview $a = 0.5, \lambda = 0.05$	0.2577 ($\pm$0.1174)
multiview $a = 0.5, \lambda = 0.0005$	0.2659 ($\pm$0.1189)
multiview $a = 0.8, \lambda = 0$	0.2328 ($\pm$0.1129)
multiview $a = 0.8, \lambda = 0.05$	0.2251 ($\pm$0.1172)
multiview $a = 0.8, \lambda = 0.0005$	0.2255 ($\pm$0.1165)
multiview $a = 1, \lambda = 0$	0.2006 ($\pm$0.1165)
multiview $a = 1, \lambda = 0.05$	0.1989 ($\pm$0.1145)
multiview $a = 1, \lambda = 0.0005$	0.1989 ($\pm$0.1132)

proportionate linearly both to map area and to N, thus method scalability is bounded by the number of UAVs and the size of the flight area/3D map.

5. Conclusions

In this paper, a centralized, multiview method for robust, on-line, on-the-fly 3D localization and mapping of human crowds in known, large-scale outdoor environments was presented, assuming their independent visual detection on the camera feed of multiple UAVs. The proposed method aims at enhancing vision-assisted human crowd avoidance, in line with common UAV safety regulations, since the resulting 3D crowd annotations may be employed for on-line mission/path replanning during deployment of a UAV fleet. The method relies on: a) an elaborate engineering pipeline, and b) novel algorithms for Bayesian fusion of multiview data and for identification of large, flat terrain areas. Thus, it also outputs potential UAV landing sites as a useful by-product. Evaluation on both synthetic and real world drone video sequences reliably indicates the superiority of the proposed method

in comparison to the naive alternative, which does not employ our novel algorithms.

Acknowledgements

The research leading to these results has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement number 731667 (MULTIDRONE). This publication reflects only the authors views. The European Union is not liable for any use that may be made of the information contained therein.

References

[1] I. Karakostas, I. Mademlis, N. Nikolaidis, I. Pitas, Shot type constraints in UAV cinematography for autonomous target tracking, Information Sciences 506 (2020) 273–294.

[2] I. Karakostas, I. Mademlis, N. Nikolaidis, I. Pitas, UAV cinematography constraints imposed by visual target tracking, in: Proceedings of the IEEE International Conference on Image Processing (ICIP), 2018.

[3] R. Cunha, M. Malaca, V. Sampaio, B. Guerreiro, P. Nousi, I. Mademlis, A. Tefas, I. Pitas, Gimbal control for vision-based target tracking, in: European Signal Processing Conference, Satellite Workshop (EUSIPCOW), 2019.

[4] P. Nousi, I. Mademlis, I. Karakostas, A. Tefas, I. Pitas, Embedded UAV real-time visual object detection and tracking, in: Proceedings of the IEEE International Conference on Real-time Computing and Robotics (RCAR), 2019.

[5] I. Mademlis, V. Mygdalis, N. Nikolaidis, I. Pitas, Challenges in autonomous UAV cinematography: an overview, in: Proceedings of the IEEE International Conference on Multimedia and Expo (ICME), 2018.

[6] I. Mademlis, N. Nikolaidis, A. Tefas, I. Pitas, T. Wagner, A. Messina, Autonomous unmanned aerial vehicles filming in dynamic unstructured outdoor environments, IEEE Signal Processing Magazine 36 (2018) 147–153.

[7] I. Mademlis, V. Mygdalis, N. Nikolaidis, M. Montagnuolo, F. Negro, A. Messina, I. Pitas, High-level multiple-UAV cinematography tools for covering outdoor events, IEEE Transactions on Broadcasting 65 (2019) 627–635.

[8] I. Mademlis, N. Nikolaidis, A. Tefas, I. Pitas, T. Wagner, A. Messina, Autonomous UAV cinematography: A tutorial and a formalized shot type taxonomy, ACM Computing Surveys 52 (2019) 105.

[9] C. Symeonidis, I. Mademlis, N. Nikolaidis, I. Pitas, Improving neural Non-Maximum Suppression for object detection by exploiting interest-point detectors, in: Proceedings of the IEEE International Workshop on Machine Learning for Signal Processing (MLSP), 2019.

[10] M. Tzelepi, A. Tefas, Human crowd detection for drone flight safety using Convolutional Neural Networks, in: Proceedings of the European Signal Processing Conference (EUSIPCO), IEEE, 2017.

[11] A. Hornung, K. M. Wurm, M. Bennewitz, C. Stachniss, W. Burgard, Octomap: An efficient probabilistic 3d mapping framework based on octrees, Autonomous Robots 34 (2013) 189–206.

[12] A. Torres-González, J. Capitán, R. Cunha, A. Ollero, I. Mademlis, A multidrone approach for autonomous cinematography planning, in: Proceedings of the Iberian Robotics Conference, 2017.

[13] D. F. Maune, Digital elevation model technologies and applications: the DEM users manual, ASPRS Publications, 2007.

[14] W. Li, J. Gu, B. Chen, J. Han, Incremental instance-oriented 3D semantic mapping via rgb-d cameras for unknown indoor scene, Discrete Dynamics in Nature and Society 2020 (2020).

[15] M. Grinvald, F. Furrer, T. Novkovic, J. J. Chung, C. Cadena, R. Siegwart, J. Nieto, Volumetric instance-aware semantic mapping and 3D object discovery, IEEE Robotics and Automation Letters 4 (2019) 3037–3044.

[16] E. Kakaletsis, N. Nikolaidis, Potential UAV landing sites detection through Digital Elevation Models analysis, in: Proceedings of the European Signal Processing Conference, Satellite Workshop (EUSIPCOW), 2019.

[17] H. Oleynikova, Z. Taylor, M. Fehr, R. Siegwart, J. Nieto, Voxblox: Incremental 3D Euclidean signed distance fields for on-board MAV planning, in: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2017.

[18] C. L. Miller, R. A. Laflamme, The Digital Terrain Model-: Theory & Application, MIT Photogrammetry Laboratory, 1958.

[19] R. Weibel, M. Heller, Digital terrain modelling, Oxford University Press, 1993.

[20] G. Priestnall, J. Jaafar, A. Duncan, Extracting urban features from LiDAR digital surface models, Computers, Environment and Urban Systems 24 (2000) 65–78.

[21] L. Zhang, Automatic Digital Surface Model (DSM) Generation from Linear Array Images, Institut für Geodäsie und Photogrammetrie Zürich: Mitteilungen, Institute of Geodesy and Photogrammetry, 2005.

[22] E. Remolina, B. Kuipers, Towards a general theory of topological maps, Artificial Intelligence 152 (2004) 47–104.

[23] H. Zender, O. M. Mozos, P. Jensfelt, G.-J. Kruijff, W. Burgard, Conceptual spatial representations for indoor mobile robots, Robotics and Autonomous Systems 56 (2008) 493–502.

[24] S. Friedman, H. Pasula, D. Fox, Voronoi Random Fields: extracting topological structure of indoor environments via place labeling, in: Proceedings of the International Joint Conferences on Artificial Intelligence (IJCAI), 2007.

[25] R. Polastro, F. Corrêa, F. Cozman, J. Okamoto, Semantic mapping with a probabilistic description logic, in: Proceedings of the Brazilian Symposium on Artificial Intelligence, Springer, 2010.

[26] A. Anand, H. S. Koppula, T. Joachims, A. Saxena, Contextually guided semantic labeling and search for three-dimensional point clouds, The International Journal of Robotics Research 32 (2013) 19–34.

[27] D. Pangercic, B. Pitzer, M. Tenorth, M. Beetz, Semantic object maps for robotic housework-representation, acquisition and use, in: Proceedings

of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2012.

[28] A. Pronobis, P. Jensfelt, Large-scale semantic mapping and reasoning with heterogeneous modalities, in: Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), 2012.

[29] R. de Nijs, S. Ramos, G. Roig, X. Boix, L. Van Gool, K. Kühnlenz, On-line semantic perception using uncertainty, in: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2012.

[30] N. Mitsou, R. de Nijs, D. Lenz, J. Frimberger, D. Wollherr, K. Kühnlenz, C. Tzafestas, Online semantic mapping of urban environments, in: Proceedings of the International Conference on Spatial Cognition, Springer, 2012, pp. 54–73.

[31] L. Boominathan, S. Kruthiventi, R. Venkatesh Babu, Crowdnet: a deep convolutional network for dense crowd counting, in: Proceedings of the ACM Multimedia Conference, ACM, 2016.

[32] J. Shao, K. Kang, C. Change Loy, X. Wang, Deeply learned attributes for crowded scene understanding, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015.

[33] D. Babu Sam, S. Surya, R. Venkatesh Babu, Switching convolutional neural network for crowd counting, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017.

[34] G. Castellano, C. Castiello, C. Mencar, G. Vessio, Crowd detection for drone safe landing through Fully-Convolutional Neural Networks, in: Proceedings of the International Conference on Current Trends in Theory and Practice of Informatics, Springer, 2020.

[35] Q. Wang, M. Chen, F. Nie, X. Li, Detecting coherent groups in crowd scenes by multiview clustering, IEEE Transactions on Pattern Analysis and Machine Intelligence 42 (2018) 46–58.

[36] X. Liu, Multi-view 3D human tracking in crowded scenes, in: Proceedings of the AAAI Conference on Artificial Intelligence, 2016.

[37] W. Ge, R. T. Collins, Crowd detection with a multiview sampler, in: Proceedings of the European Conference on Computer Vision (ECCV), Springer, 2010.

[38] M. Voit, K. Nickel, R. Stiefelhagen, A Bayesian approach for multi-view head pose estimation, in: Proceedings of the IEEE International Conference on Multi-sensor Fusion and Integration for Intelligent Systems, 2006.

[39] P. Peng, Y. Tian, Y. Wang, J. Li, T. Huang, Robust multiple cameras pedestrian detection with multi-view Bayesian network, Pattern Recognition 48 (2015) 1760–1772.

[40] L. Rybok, M. Voit, H. K. Ekenel, R. Stiefelhagen, Multi-view based estimation of human upper-body orientation, in: Proceedings of the IEEE International Conference on Pattern Recognition (ICPR), 2010.

[41] M. Garg, A. Kumar, P. Sujit, Terrain-based landing site selection and path planning for fixed-wing uavs, in: Proceedings of the IEEE International Conference on Unmanned Aircraft Systems (ICUAS), 2015.

[42] M. Aydin, E. Kugu, Finding smoothness area on the topographic maps for the unmanned aerial vehicle's landing site estimation, in: Proceedings of the IEEE International Conference on Digital Information and Communication Technology and its Applications (DICTAP), 2016.

[43] D. Maturana, S. Scherer, 3D convolutional neural networks for landing zone detection from LiDAR, in: Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), 2015.

[44] A. E. Johnson, A. R. Klumpp, J. B. Collier, A. A. Wolf, LiDAR-based hazard avoidance for safe landing on mars, Journal of Guidance, Control, and Dynamics 25 (2002) 1091–1099.

[45] S. Scherer, L. Chamberlain, S. Singh, Autonomous landing at unprepared sites by a full-scale helicopter, Robotics and Autonomous Systems 60 (2012) 1545–1562.

[46] D. H. Douglas, T. K. Peucker, Algorithms for the reduction of the number of points required to represent a digitized line or its caricature,

Cartographica: The International Journal for Geographic Information and Geovisualization 10 (1973) 112–122.

[47] A. S. Glassner, An introduction to ray tracing, Elsevier, 1989.

[48] S. D. Roth, Ray casting for modeling solids, Computer graphics and image processing 18 (1982) 109–144.

[49] M. Stone, The opinion pool, The Annals of Mathematical Statistics (1961) 1339–1342.

[50] M. Rufo, J. Martin, C. Pérez, Log-linear pool to combine prior distributions: A suggestion for a calibration-based approach, Bayesian Analysis 7 (2012) 411–438.

[51] C. C. Hau, Handbook of pattern recognition and computer vision, World Scientific, 2015.

[52] Arcmap: How slope works, http://desktop.arcgis.com/en/arcmap/10.3/tools/spatial-a 2017. Accessed: 14/12/2017.

[53] R. Duda, P. Hart, Pattern Classification and Scene Analysis, Wiley, 1973.

[54] M. Quigley, B. Gerkey, K. Conley, J. Faust, T. Foote, J. Leibs, E. Berger, R. Wheeler, A. Ng, ROS: an open-source Robot Operating System, in: Proceedings of the IEEE International Conference on Robotics and Automation (ICRA) Workshop on Open Source Robotics, 2009.

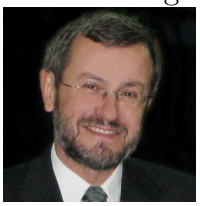
[55] E. Kakaletsis, M. Tzelepi, P. I. Kaplanoglou, C. Symeonidis, N. Nikolaidis, A. Tefas, I. Pitas, Semantic map annotation through UAV video analysis using deep learning models in ROS, in: Proceedings of the International Conference on Multimedia Modeling (MMM), Springer, 2019.

Author Biographies

Efstratios Kakaletsis received the Diploma (2010) in Electrical and Computer Engineering from the Department of Electrical and Computer Engineering at the Aristotle University of Thessaloniki, Greece and a M.Sc. degree (2014) in Computer Science - Digital Media from the Department of Informatics of the same university. Currently, he is a Ph.D. student, employed as a research assistant at the Artificial Intelligence and Information Analysis (AIIA) Laboratory. He has participated in 4 European Union-funded Research and Development projects. His current research interests include computer graphics, computer vision, and machine learning. Email: ekakalets@csd.auth.gr

Ioannnis Mademlis is a computer scientist, specialized in artificial intelligence. He received a B.Sc. (2007) and a M.Sc. degree (2010) from the University of Ioannina, Greece. Additionally, he received a M.Sc. degree in intelligent systems/cognitive science (2014) and a Ph.D. in machine learning and computer vision (2018), both from the Aristotle University of Thessaloniki, Greece (AUTH). While pursuing his Ph.D., he was granted a 1-year academic excellence scholarship. Presently, he is employed as a postdoctoral research associate at the Artificial Intelligence and Information Analysis Laboratory (AIIA) in AUTH. He has participated in 5 European Union-funded RD projects, having co-authored more than 30 publications in academic journals and international conferences. His current research interests include machine learning, computer vision, natural computing, autonomous robotics and intelligent cinematography. E-mail: imademlis@csd.auth.gr

Nikos Nikolaidis received the Diploma of Electrical Engineering and the Ph.D. degree in Electrical Engineering from the Aristotle University of Thessaloniki, Thessaloniki, Greece, in 1991 and 1997, respectively. He is currently Associate Professor at the Department of Informatics, Aristotle University of Thessaloniki. He has coauthored 1 book, 15 book chapters, 61 journal papers and 189 conference papers and co-edited one book and two special issues in journals. Moreover he has co-organized 6 special sessions in international conferences. The number of citations to his work by third authors exceeds 5300 (h-index 31, Source: Google Scholar). He has participated into 25 research projects funded by mainly by EU but also national funds. His current areas of interest include computer/robot vision, image/video processing and analysis, analysis of motion capture data, computer graphics and visualization. Dr. Nikolaidis is currently serving as associate/area editor for Signal Processing: Image Communication, EURASIP Journal on Image and Video Processing and IET Image Processing. He served as Technical Program chair of IEEE IVMSP 2013 workshop, and Publicity co-chair of EUSIPCO 2015 and IEEE ICIP 2018. Dr. Nikolaidis is a Senior Member of IEEE and member of the Technical Chamber of Greece. E-mail: nnik@csd.auth.gr



Ioannis Pitas (IEEE Distinguished Lecturer, EURASIP fellow) received the Diploma and PhD degree in Electrical Engineering, both from the Aristotle University of Thessaloniki (AUTH), Greece. Since 1994, he has been a Professor at the Department of Informatics of AUTH and Director of the Artificial Intelligence and Information Analysis (AIIA) lab. He served as a Visiting Professor at several Universities. His current interests are in the areas of computer vision, machine learning, autonomous systems, intelligent digital media, image/video processing, human-centred interfaces, affective computing, 3D imaging and biomedical imaging. He has published

over 906 papers, contributed in 47 books in his areas of interest and edited or (co-)authored another 11 books. He has also been member of the program committee of many scientific conferences and workshops. In the past he served as Associate Editor or co-Editor of 9 international journals and General or Technical Chair of 4 international conferences. He participated in 70 RD projects, primarily funded by the European Union and is/was principal investigator/researcher in 42 such projects. He has 31600+ citations to his work and h-index 85+ (Google Scholar). Prof. Pitas leads the International AI Doctoral Academy (IAIDA) of the European H2020 RD project AI4Media https://ai4media.eu/. He is chair of the Autonomous Systems Initiative: http://asi.politecnica.unige.it/. E-mail: pitas@csd.auth.gr