Real-Time Detection and Tracking of Multiple Humans from High Bird's-Eye Views in the Visual and Infrared Spectrum

Julius Kümmerle, Timo Hinzmann,
Anurag Sai Vempati and Roland Siegwart

Autonomous Systems Lab, ETH Zurich

Abstract. We propose a real-time system to detect and track multiple humans from high bird's-eye views. First, we present a fast pipeline to detect humans observed from large distances by efficiently fusing information from a visual and infrared spectrum camera. The main contribution of our work is a new tracking approach. Its novelty lies in online learning of an objectness model which is used for updating a Kalman filter. We show that an adaptive objectness model outperforms a fixed model. Our system achieves a mean tracking loop time of 0.8 ms per human on a 2 GHz CPU which makes real time tracking of multiple humans possible.

1 Introduction

Research on fully autonomous Unmanned Aerial Vehicles (UAVs) became very popular with the development of powerful but lightweight computers and sensors. Besides the enormous popularity of multi-copters, fixed-wing UAVs are also becoming steadily more interesting mainly due to their large flight range. In a classical scenario they are used to first localize victims and then send their GPS locations to a rescue team on the ground. In this context, autonomous human tracking from fixed-wing UAVs faces several challenges. The main challenge is the flight altitude of around 50-150 m. High resolution imagery is needed to provide enough detail for accurately detecting humans. Searching small objects in large images is computationally expensive which makes the human search with real-time constraint challenging, especially on UAVs with limited computational power. For that reason, we use a long wave thermal camera (FLIR Tau 2 19 mm). In many scenes humans significantly differ in temperature from most of the background. This can be used to reduce the detection search space and thereby increase efficiency of the human detection. The main drawback of our thermal camera is the relatively low resolution of 640×512. We use a visual spectrum camera with a high resolution of 1600×1200 alongside the thermal camera to get more detail.

2 Related work

Object tracking is one of the big topics of computer vision and many different approaches have been developed in the last decades for which [1] gives an excellent overview. Just recently, different research groups started to use an objectness measure in trackers [2, 3]. The concept of objectness is proposed by [4] and essentially aims at deciding whether an image patch contains an object or not before searching for specific object classes. Cheng et al. [5] developed an efficient algorithm for calculating an objectness measure, denoted by *BING objectness*, and thereby enabled the use of objectness in real time applications. Objectness is shown to increase accuracy when used as an additional score for tracking [2, 3]. Liang et al. [3] initialize a tracker with an adaptive objectness model which is learned by an adaptive SVM [6] in the frame of the first detection. The model combines generic objectness with object specific information. The integration of the adaptive objectness model into state-of-the-art trackers is reported to outperform the original trackers in most cases [3]. Our tracker approach is also based on an adaptive objectness model. The difference to [3] lies in an ongoing online learning process to gradually transform the generic objectness model to a discriminative object specific model. The online learning framework is based on the *Passive-Aggressive* algorithm introduced in [7]. The objectness detection is integrated into a Kalman filter [8, 9]. For human detection we make use of the popular Histogram of Orientated Gradients (HOG) features [10].

3 Approach

In the following, we describe the approach used for human detection and tracking.

3.1 Detection pipeline

Human detection from heights of 50-150 m brings along many challenges. Humans only occupy few pixels of the image which makes a detailed search necessary and thereby makes the detection particularly computationally expensive. To speed up the search for humans we use the long wave infrared spectrum as additional information. Fig. 1 shows an example where humans clearly stand out from the background. Compared to the corresponding visual spectrum image the infrared image shows significantly less texture. This allows us to use a simple but fast blob detector in the infrared image. At each blob we define centered bounding boxes of different sizes. These are the candidates which are given to a HOG detector. The linear SVM is trained on 2400 positive and 24000 negative human samples extracted from four infrared datasets [11–14]. The HOG detector is designed to classify patches with heights of 6-36 pixels. Compared to the original HOG detector [10] the humans are significantly smaller so we had to find new suitable parameters as described in Sec. 4.1.

(a) (b)

Fig. 1. (a) Crop of infrared image showing humans with high contrast from background. (b) Corresponding crop of visual spectrum image.

To increase accuracy, we also use the high resolution visual spectrum image. The challenge in fusing information from multiple cameras is to find corresponding areas. Standard appearance based matching algorithms [15] can only be applied if the appearance in the images is similar. This cannot be guaranteed when using a visual and an infrared spectrum camera. Some parts of an object might show high contrast in the infrared spectrum whereas they may not be distinguishable from the background in the visual spectrum. Moreover, an object in the visual spectrum often shows significant texture which cannot be perceived in the infrared spectrum. To circumvent the problem of lacking similarity in appearance we use the assumption that the distance to the object is large compared to the baseline of the cameras. For the theoretical case that the cameras have coincident optical centers and their principle axes are aligned the matching problem is trivially solved by finding pixels with the same projection ray. For a vertically aligned stereo pair with parallel principle axes and no distortion, the pixel error err_{pix} introduced by a non-vanishing baseline b can be calculated as

$$err_{pix} = \frac{f}{l_{pix}} \cdot \frac{b}{d}, \tag{1}$$

where f is the focal length, l_{pix} denotes the size of a pixel and d is the depth of the targeted 3D point P as illustrated in Fig. 2. f and l_{pix} belong to the camera the error is attributed to. From this we can calculate the minimal depth d_{min} of a 3D point to make a sub-pixel error w.r.t the infrared camera ($f_{ir} = 19\,\text{mm}$, $l_{pix,ir} = 17\,\mu\text{m}$, $b = 30\,\text{mm}$)

$$d_{min} = \frac{f}{l_{pix}} \cdot b \approx 34\,\text{m} \tag{2}$$

Since we target human detection at longer ranges, the error due to baseline and scene depth variations can be neglected.

For the candidates which were classified as humans in the infrared spectrum we calculate the corresponding bounding boxes in the visual spectrum image.

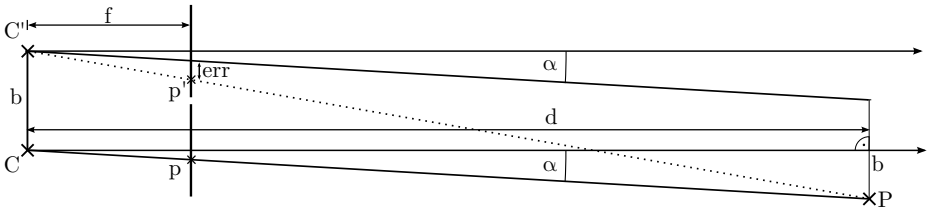


Fig. 2. Stereo camera pair with optical centers C and C'. The principle axes are parallel. With the known projection p of the 3D point P, the corresponding projection p' in the other camera is approximated by assuming a vanishing baseline. The resulting error is denoted as *err*.

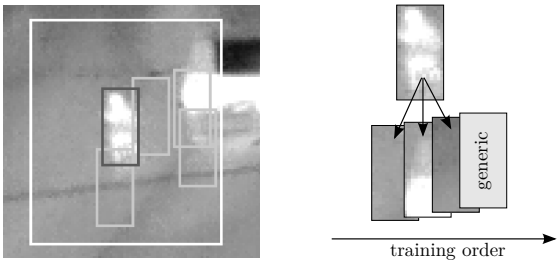
To compensate errors resulting from a weak stereo calibration (calibration of infrared cameras is comparably hard because of blurry imagery [16]) we extend the number of candidates in the visual spectrum image by sampling around the correspondences. As a final stage of our pipeline, we classify these candidates by another HOG detector which was trained on 3000 humans from bird's-eye views and 25000 negative samples.

3.2 Tracker

We perform the tracking of humans on the infrared images because in many scenes humans are more salient in the infrared than in the visual spectrum. Furthermore, we know the position of a detected person more accurately in the infrared than in the visual spectrum due to the weak camera calibration.

The key idea of our tracker approach is to use the BING algorithm [5] to rapidly generate a small number of candidates for redetecting the person in the new frame. The BING algorithm essentially is a linear SVM which approximates the SVM model and feature vector by sums of binary vectors to achieve high efficiency. A feature vector is constructed by calculating the absolute gradient field of the target region, resizing the field to 8×8 and interpreting the result as a 64 D vector. Because of the resizing to 8×8 usually the model is underfitting when trained on a specific object class and therefore leads to limited usability as a single object class classifier on a large region. Therefore, we apply the BING algorithm only in a local area around the last tracker position as shown in Fig. 3(a).

Originally, the BING algorithm is used with a fixed objectness model generated by training on multiple object classes which is useful for detecting general objects. For tracking, the object is the same and might only change appearance in small ranges. Therefore, we want to learn object specific information and include it into the model. The goal of the learning process is to bias the model towards the object without completely loosing the generic objectness information which is valuable to prevent drift. We use a soft margin version of the *Passive-Aggressive* (PA) algorithm [7] to adjust the model while tracking. We define three different types of training data. *Negative* training data is generated



(a) (b)

Fig. 3. (a) BING candidates (gray bounding boxes) in local area (white box). The highest scoring candidate (dark bounding box) is well fitting the tracked person. (b) Training order with one object specific positive, three negatives and the generic objectness model.

by randomly sampling the neighborhood. An *object specific positive* sample is a candidate which is identified as the tracked object based on appearance and motion. We define the third type of sample as *generic positive*. The original objectness model serves as a generic positive sample. Training is only performed when an object specific positive sample is identified. We sample three negatives and train on the same object specific positive after each negative sample. After these six training rounds we train on the generic objectness model as illustrated in Fig. 3(b). This procedure ensures a balanced ratio of positives to negatives.

The identification of a good candidate is essential for the learning procedure. Since errors in the identification cannot be eliminated in practice we use a low learning rate to be more robust to false positives. Furthermore, continuous training on the generic objectness model partly compensates false updates. As mentioned before, we use an appearance model and a motion model for the identification of a positive candidate. To capture the appearance of the object we simply store positive identified patches in a FIFO storage with a capacity of five patches. To check if a candidate agrees with the stored appearance we use the $\mathbf{L}_2$-norm of the difference of the candidate to each stored patch and check if the smallest norm is lower than a dynamic threshold. The threshold decreases with the number of successive positive identified candidates and thereby expresses the trust in the appearance model as well as in the BING model. If multiple candidates are classified as positive then we use the distance to the predicted position from the constant velocity motion model to choose a candidate. We take this uncertainty into account by weighting the learning rate of the learning algorithm by the inverse of the number of identified candidates. We have chosen this relatively simple method to compare appearance of patches since many sophisticated methods use descriptors that are not reliable on textureless and small objects as in our case. If the objects show more detail, the identifier step can be easily replaced with other methods while keeping the learning process the same.

To fuse information of the BING detector, motion model and human detection pipeline we use a Kalman filter in which the BING detector represents one update step which is weighted with the trust in the model. The BING detection is only used to update position whereas the size is updated by the detection pipeline. This speeds up the BING detection since the search is limited to a single size.

4 Experimental Results

For the testing of the detection pipeline and the tracker we collected a new dataset with our stereo camera pair. The videos show city scenes with a camera angle of around 45 ° to the horizontal so that most of the humans' front view profiles are still visible. Humans have distances in the range of 30 m-80 m to the cameras whereas the mean distance is around 50 m. This results in human heights in the infrared images of 6-36 pixels whereas the mean is around 16 pixels. In the visual spectrum humans have roughly double the pixel heights as shown in Fig. 4. Humans smaller than 12 infrared pixels often have low contrast in the infrared image and also lose the typical human shape in both the infrared and visual spectrum images. Besides the small sizes of the humans, partial occlusions and groups of humans make the dataset challenging. Furthermore, the stereo camera pair is moved and rotated which makes the tracking significantly harder.

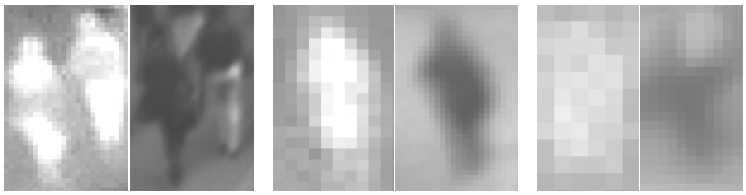


Fig. 4. Examples of humans in our dataset. (a) Humans of heights > 30 infrared pixels. (b) Human of height 15 infrared pixels. (c) Human of height < 10 infrared pixels.

4.1 Detection pipeline

For the evaluation of the detection pipeline we split the detector cascade in its three stages, which are the blob detector, the HOG detector in the infrared spectrum (HOG_is) and the HOG detector in the visual spectrum (HOG_vs). We inspect the performance of the system step by step starting with only the blob detector, then adding the HOG_is stage and finally using the whole pipeline. As a performance measure we plot miss rates over false positives per image (FPPI). The evaluation is only performed on the area which is represented in both images where Fig. 5 shows the fused images.

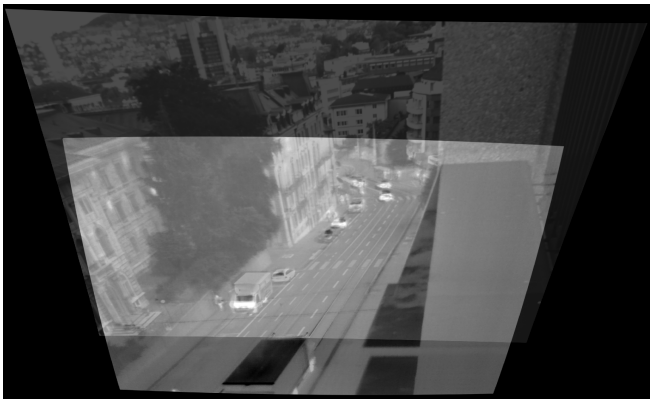


Fig. 5. Image fusion of infrared and visual spectrum image. A pixel mapping based on the assumption of coincident camera centers is used. Images are undistorted and rectified.

Blob detector The main goal of the blob detector is to reduce the number of candidates for the computationally expensive classifiers in the next stages. We set the limit of blob detections to 200 to guarantee an upper bound on the runtime. On the other hand, the blob detector has to find most of the persons so that they reach the strong classifiers in the second and third stage. In other words, the miss rate should be small. The performance of the blob detector is shown in Fig. 6. We increased the thresholding spacing to make the blob detector more tolerant to intensity changes inside a blob. Therefore, increasing the threshold spacing results in detecting more low quality blobs. As we can see from Fig. 6, the performance rapidly increases for a FPPI range of 1-30 and levels out at FPPI of 40 and a miss rate of 0.2. From the high miss rates at low FPPI we see that only few humans in the infrared really appear as a homogeneous blob. Most of the humans show temperature differences e.g. warm head and legs and relative cold upper body. Some of the humans ($\approx 20\,\%$) cannot be detected by the blob detector since they show very little contrast to the background if at all. The differences in blob quality can be seen in Fig. 1 and Fig. 4. To find most of the humans and guarantee fast runtimes, we use the blob detector configuration which results in a miss rate of $20\,\%$ with a FPPI of 40. The high number of false positives shows that a simple blob detector in the infrared spectrum is not a good human detector.

Infrared HOG detector The HOG_is shall reduce the number of false positives significantly. We aim at FPPI < 0.1 or preferably at FPPI < 0.01 for the final detector since each false positive detection initializes a wrong tracker. In the worst case the wrong trackers cannot be deleted quickly which results in accumulation of wrong trackers. Hence, low FPPI is an essential prerequisite for a high tracker performance. We achieved best results with the following HOG

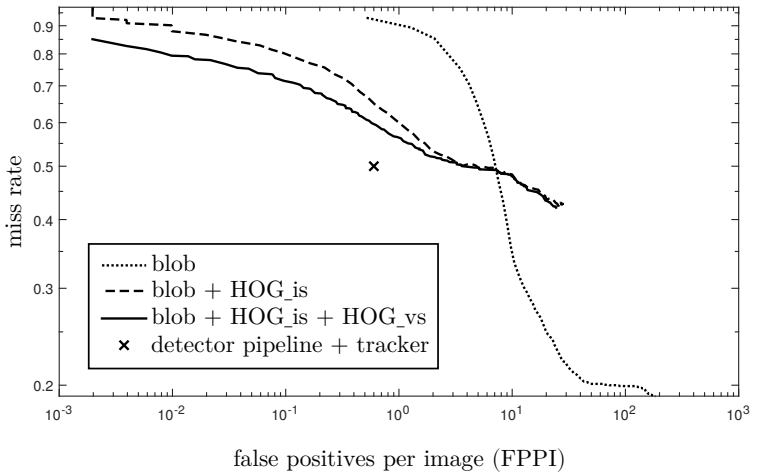


Fig. 6. Experimental performance results on our dataset. The detector shows significant decrease in FPPI over a wide range with each stage (see three curves). The combination of detector and tracker reduces the miss rate by 10% at same FPPI compared to the detector pipeline alone (see cross).

descriptor configuration: patch size = 16×32, block size = 8×8, block shift = 2, cell size = 4×4 and bin number = 9. Two retraining rounds on hard false positives gave an improvement in miss rate of 5 % at FPPI = 0.1. Our best performing HOG_is for FPPI of 0.01-0.1 is shown in Fig. 6. The graph shows that adding the HOG_is stage to the blob detector significantly improves performance in the low FPPI range which cannot even be reached by the blob detector. At FPPI = 0.01 and FPPI = 0.1 the miss rate is 0.87 and 0.80, respectively. The performance is in the same range as in state-of-the-art pedestrian detection of heights 30-80 pixels on visual spectrum images [17].

Visual spectrum HOG detector The final HOG detection in the visual spectrum is expected to further reduce the FPPI at the same miss rate since some objects like warm windows or wheels might appear similar to humans in the infrared image whereas in the visual spectrum these objects clearly appear as non-humans. Since our stereo pair is not synchronized and the calibration is weak we sample multiple candidates in the visual spectrum image around the mapped detection from the infrared image. As presented in Fig. 7, the number of samples and the spacing influences the overall performance. If the sampling cannot compensate for the mapping errors the HOG_vs stage does not give any improvement. If the samples compensate for the mapping errors the performance is increased significantly by the HOG_vs stage. The best performance of the complete detection pipeline at FPPI = 0.01 and FPPI = 0.1 is reached for the

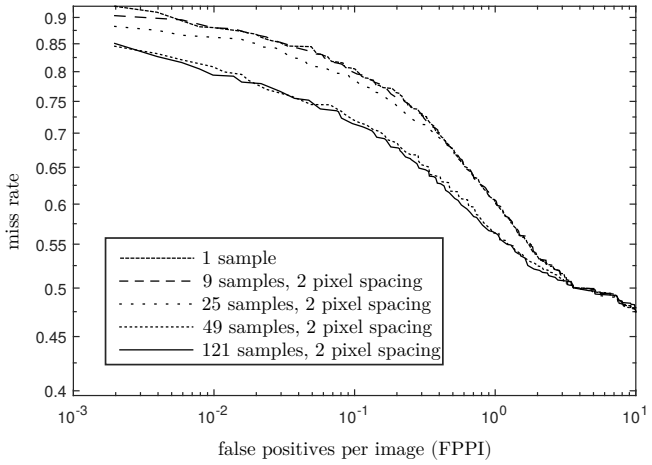


Fig. 7. Experimental performance results of the detector pipeline for different sampling settings in the visual spectrum. The pixel spacing is given in visual spectrum image pixels. The sampling highly influences the effectiveness of the HOG detector in the visual spectrum. With 49 samples most of the mapping errors are compensated and a reduction of up to 10% in miss rate is achieved by the classifier in the visual spectrum.

setting of the HOG_is with FPPI = 0.22/miss rate = 0.76 and FPPI = 0.45/miss rate = 0.68, respectively. Fig. 6 shows that with image fusion an improvement of almost 10% in miss rate in the relevant FPPI range can be achieved.

4.2 Tracker

As explained in Sec. 3.2 the key idea of the tracker is the adaptive BING model which is adjusted while tracking. We want to verify that learning object specific information is beneficial for tracking and outperforms a fixed model. To evaluate the performance of the BING detector independently from the rest of the tracker we perform the following experiment. First we use the original objectness model and run the BING algorithm in local areas around the annotated person positions. We take the five best candidates and check for overlaps with the ground truth human bounding box of more than 80 %. The candidate with the highest overlap is used to calculate an intuitive performance score

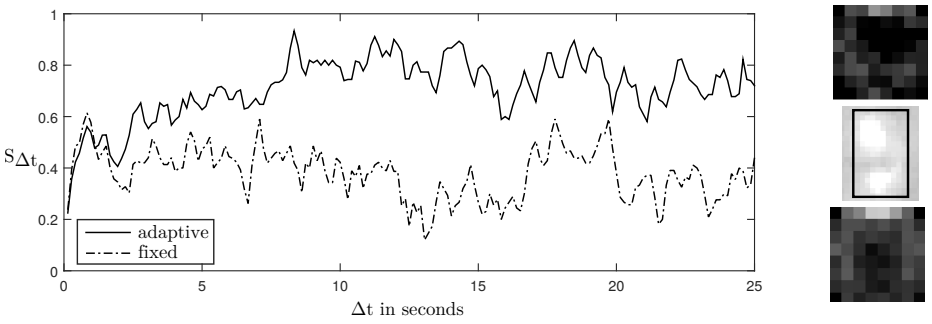
$$s_{a,\Delta t} = \begin{cases} \frac{6-r_{a,\Delta t}}{5} & \text{if highest overlap} > 80\% \\ 0 & \text{else} \end{cases} \tag{3}$$

where a denotes the annotation, Δt is the time passed since the first frame of the annotation and $r \in [1, 5]$ denotes the ranking based on the classification score of the BING detector for the candidate with the highest overlap of more than 80 %. For example, if the candidate with the second highest classification score

($r = 2$) has the highest overlap with the annotation and the overlap is over 80 % then the performance score is 4/5. If none of the five candidates has an overlap of more than 80 % then the performance score is set to zero. The performance over all annotations is measured with the normed cumulative score

$$S_{\Delta t} = \frac{\sum_a s_{a,\Delta t}}{n_{\Delta t}}, \tag{4}$$

where $n_{\Delta t}$ is the number of humans annotated at Δt. To evaluate the BING detector with an online learned model we use the same scoring system but relearn the model with the highest overlapping candidate if the overlap is more than 80 %. This simulates a learning procedure with a robust identification step. The results for our infrared dataset are shown in Fig. 8(a). We notice that after a



(a) (b)

Fig. 8. (a) Performance of BING detector in the infrared spectrum with fixed and adaptive model measured with the cumulative score (see equation 4). The adaptive model outperforms the fixed model after few seconds. (b) Example of adaptive model (top) and fixed model (bottom) for the same tracked object (middle). The 64D model vectors (SVM weights) are represented as 8×8 patches. Bright means high and dark means low value.

few training rounds the BING detector with an adaptive model outperforms the detector with a fixed model. The score curves show a mean difference of 0.3 which means that on average the BING detector with adaptive model needs 3 candidates less to find the tracked object than the detector with the fixed generic model. Fig. 8(b) gives an example of a fixed and a learned model for a tracked human. The human clearly appears as a big blob at the top and a smaller blob at the bottom. The original BING objectness model assumes an object to have a single closed boundary which is not suitable for this example. The adaptive model shows the two blob characteristic and their size difference and thereby outperforms the fixed model.

For the evaluation of our complete detection-tracking framework we use a detection pipeline with FPPI = 0.10 and miss rate = 0.72. The complete system

significantly outperforms the detection pipeline with a miss rate = 0.50 and FPPI = 0.62 as shown in Fig. 6. For the same miss rate the detector has a FPPI of 5.0 which is 8 times more than the FPPI of the complete system.

4.3 Runtime

Table 1 shows runtime results for the detector and tracker. The experiments were run on a 2.0 GHz CPU. The detector pipeline is the time consuming part of the

stage	mean runtime in ms
blob detector	16.3
HOG_is	22.0
HOG_vs	2.3
complete detection pipeline	40.6
BING detection	0.17
PA learning	0.08
identification	0.46
other	0.10
complete tracker loop	0.81

Table 1. Experimental runtime results of detector (top row) and tracker (bottom row). Detection runtime is 50 times longer than tracker loop time.

system. The HOG_is stage has to classify significantly more candidates than the HOG_vs stage which explains the difference in runtime. Usually, the detection pipeline is not used on every frame to save computational power. The efficient reduction of candidates by the BING detector results in very fast tracker loop times which enables tracking of many humans simultaneously.

5 Conclusion and Future Work

We introduced a simple fusion technique for infrared and visual spectrum imagery which we use in our detection pipeline. We showed from experimental results that our detection pipeline significantly improves performance by using classifiers in both the infrared and visual spectrum. For tracking humans in the infrared spectrum we proposed a novel tracking approach that is based on an adaptive objectness detector which is learned while tracking. Our experiment shows that learning biases the objectness model to a specific object and thereby leads to more meaningful candidate ranking. In future work, we want to fuse objectness proposals in both the infrared and visual spectrum to arrive at an even more reliable ranking. Furthermore, we want to combine objectness with other more sophisticated online learning algorithms and explore other methods for the identification of the tracked object. We also plan to evaluate our system on UAV datasets to determine the performance of human detection and tracking at higher altitudes and camera speeds.

ACKNOWLEDGMENT

The research leading to these results has received funding from the European Commission's Seventh Framework Programme (FP7/2007-2013) under grant agreement n°600958 (SHERPA).

References

1. Yilmaz, A., Javed, O., Shah, M.: Object tracking: A survey. Acm computing surveys (CSUR) **38** (2006) 13
2. Stalder, S., Grabner, H., Van Gool, L.: Dynamic objectness for adaptive tracking. In: Asian Conference on Computer Vision, Springer (2012) 43–56
3. Liang, P., Pang, Y., Liao, C., Mei, X., Ling, H.: Adaptive objectness for object tracking. (2015)
4. Alexe, B., Deselaers, T., Ferrari, V.: What is an object? In: 2010 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE (2010) 73–80
5. Cheng, M.M., Zhang, Z., Lin, W.Y., Torr, P.: Bing: Binarized normed gradients for objectness estimation at 300fps. In: Proceedings of the IEEE conference on computer vision and pattern recognition. (2014) 3286–3293
6. Yang, J., Yan, R., Hauptmann, A.G.: Adapting svm classifiers to data with shifted distributions. In: Seventh IEEE International Conference on Data Mining Workshops (ICDMW 2007), IEEE (2007) 69–76
7. Crammer, K., Dekel, O., Keshet, J., Shalev-Shwartz, S., Singer, Y.: Online passive-aggressive algorithms. Journal of Machine Learning Research **7** (2006) 551–585
8. Kalman, R.E.: A new approach to linear filtering and prediction problems. Journal of basic Engineering **82** (1960) 35–45
9. Cuevas, E.V., Zaldivar, D., Rojas, R.: Kalman filter for vision tracking. (2005)
10. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05). Volume 1., IEEE (2005) 886–893
11. Wu, Z., Fuller, N., Theriault, D., Betke, M.: A thermal infrared video benchmark for visual analysis. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops. (2014) 201–208
12. Davis, J.W., Keck, M.A.: A two-stage template approach to person detection in thermal imagery. WACV/MOTION **5** (2005) 364–369
13. Davis, J.W., Sharma, V.: Background-subtraction using contour-based fusion of thermal and visible imagery. Computer Vision and Image Understanding **106** (2007) 162–182
14. Vempati, A.S., Agamennoni, G., Stastny, T., Siegwart, R.: Victim detection from a fixed-wing uav: Experimental results. In: International Symposium on Visual Computing, Springer (2015) 432–443
15. Scharstein, D., Szeliski, R.: A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. International journal of computer vision **47** (2002) 7–42
16. Vidas, S., Lakemond, R., Denman, S., Fookes, C., Sridharan, S., Wark, T.: A mask-based approach for the geometric calibration of thermal-infrared cameras. IEEE Transactions on Instrumentation and Measurement **61** (2012) 1625–1635
17. Dollar, P., Wojek, C., Schiele, B., Perona, P.: Pedestrian detection: An evaluation of the state of the art. IEEE Transactions on Pattern Analysis and Machine Intelligence **34** (2012) 743–761