2950 Niles Road, St. Joseph, MI 49085-9659, USA
269.429.0300 fax 269.429.3852 hq@asabe.org www.asabe.org

An ASABE Meeting Presentation
DOI: 10.13031/aim.201701288
Paper Number: 1701288

Detection and localization of birds for Bird Deterrence using UAS

Shivam Goel*, Santosh Bhusal**, Matthew Taylor*, Manoj Karkee**
** School of Electrical Engineering and Computer Science,*
Washington State University
***Department of Biological System Engineering, Center for Precision and Automated Agricultural Systems, Washington State University*

Written for presentation at the
2017 ASABE Annual International Meeting
Sponsored by ASABE
Spokane, Washington
July 16-19, 2017

***ABSTRACT** Cherry, grape, and blueberry growers lose around 80 million dollars annually to bird damage in the state of Washington alone. Growers of a wide range of crops have a critical need for a safe and cost-effective method for persistent bird deterrence, which would lead to significantly reduced production costs. The goal of this research is to build a completely autonomous Unmanned Aerial System (UAS) to deter birds from the blueberry fields and grape vineyards. In the effort to build the UAS, the most vital part of its implementation is the vision system. The primary objective of this paper is to build a system to detect and localize birds. To detect birds, background subtraction algorithms have been used and the performance of various background subtraction algorithms are measured. It is found out that ViBe, a background subtraction algorithm, performs best in the bird detection scenario and provides an accuracy of 63%. In the quest of improving the bird detection speed and obtaining it in real time, a split window technique is used to improve the detection speed by 13%. To estimate the distance of the detected bird, a stereo vision system is proposed. With our current system, an accurate measure of the distance of the object is possible from 2 to 7 meters with an error accuracy of 30 centimeters. The long-term goal is to combine the efforts of the paper to successfully create a completely autonomous Smart Scarecrow that can safely, effectively and reliably scare and deter birds from high-value crops.*

***Keywords.** Bird Detection, Bird Deterrence, Bird Tracking, Bird Counting, Vineyards, Background Subtraction, Bird Deterrence, Stereo Vision System, Unmanned Aerial System*

Introduction

Cherry, grape, Honeycrisp apple, and blueberry growers lose around 80 million dollars annually to bird damage in Washington State alone (Anderson et al., 2013). Growers of a wide range of crops have a critical need for a safe and cost-effective method for persistent bird deterrence, which might lead to significantly reduced production costs. Our long-term objective is to engineer a *Smart Scarecrow*, an autonomous aerial system (UAS), to deter birds away from high-valued fruit. The approach is to intercept birds before they enter fields, as it may be difficult to scare them away after they have landed.

Towards this end, one of the most vital part is the vision system. The vision system will help in the automated detection and thereby determine the location of birds, relative to the field, which will be used to guide UAVs towards the birds. The distance estimation is the most vital part in the successful working of the autonomous system. UAVs available in the market have a very short battery span and the flight time is limited to 10-25 minutes. Due to the short flight times, a drone cannot patrol indefinitely; UAVs need to be directed only when the birds attempt to enter a protected area.

Object distance estimation is a problem that is of great relevance in computer vision and significant work has focused on acquiring distance range data (Jarvis, 1993). A wide variety of technologies has been used to estimate the distances of the objects, including LIDAR (Titan et al., 2017), RADAR (Larkin and Eisenberg, 1978). However, these methods are not as cost effective for the bird deterrence approach (DeTech Inc., 2016). Stereo vision has also been used to estimate range data of the objects (Nedevschi et al., 2004). Range detection and disparity map generation from a stereo vision system is well documented for relative short distances. However, in the Smart Scarecrow, the birds to be detected are quite far away. As a first, critical, step towards this goal, this paper uses background subtraction to separate moving objects (birds) from the static background, allowing stereo vision methods to estimate their distance (and relative location).

The paper has been organized as follows. The background and related work are discussed in the subsequent section. The section after that discusses the proposed method for bird deterrence. The next section describes the bird detection algorithms, whose results are discussed in the Results and Discussion section. The Distance Estimation section describes the algorithm used to estimate the distance of the bird. Finally, the conclusion section concludes the paper with a brief discussion of future work.

Background and Related Work

Under many circumstances, bird scaring techniques can reduce damage in crops (Ruelle P. and Bruggers, 1982). The use of scarecrows to deter grain-eating pests has provided variable success. They are one of the oldest designs of bird deterrence, but are often ineffective once birds adapt to the scarecrow (Marsh et al., 1992). Nets are used to cover crops, which prevent birds from feeding on high-value crops like cherries, blueberries, grapes (Bishop et al., 2003). However, this approach is sometimes harmful to the birds that become entangled in the nets and die, it is expensive, and can hampers the daily routine maintenance of the crop.

In recent times, drones have become very popular in agriculture (Trippochio et al., 2015). Robotics has thus found many applications in the field of agriculture. Examples include detecting soil moisture (Davidson, Malcolm WJ, et al., 2000) and navigation in Orchard (Bac, C. Wouter, et al., 2014). Grimm et al. worked on a UAV package which had the capability of autonomous flight. In this work, the UAV was designed so that a vineyard employee with minimal training can fly the aircraft. The work also surveyed various vineyards and found out that the vineyard owners were open to using new methods to deter birds which were humane. A similar work was also done by the researchers at the California State University, Bakersfield. The purpose of their UAV was to detect the flock of birds using a wireless ground sensor network and this data was sent through a cloud-based decision system to direct a hexacopter armed with a sprayer and auditory scaring devices to irritate the birds to get away from the crops (Ampatzidis et al., 2015). However, the method is still in the beginning stages of development and no significant work has been done to accurately estimate the position of the birds as the system was based upon using a simulation of bird detection.

Autonomous System for Bird Deterrence

The method of bird deterrence proposed in this paper uses drones to scare away birds. The proposed method is described in Figure 1. The diagrammatic representation shows the placement of the ground cameras for bird detection and distance estimation. A central computer will be installed connecting the cameras and the drone via Robot Operating System (ROS) (Garage and Willow, 2012). When birds approach a boundary, one or more drones will be activated. After launched, a drone will either fly to intercept a bird or start patrolling a defined boundary to help scare the birds away from the field. The drones have a relatively short flight time (Anderson and Gaston, 2013), depending upon the battery used they last from 15-30 minutes. Due to this restriction, the drones cannot patrol the field all day long. The drone will return to the charging station once the birds are chased away from the boundary. The proposed method is completely autonomous and does not require any human intervention after its installation.

A somewhat similar method is being investigated by a company in the Netherlands. They are trying to create a “robobird” that mimic predator birds. Although no studies have been published, Clear Flight Solutions claim a 75% reduction in the birds visited a landfill protected by a robobird (Vanhemert, 2014). In contrast to the robobird, our method does not require human operators. To achieve full autonomy, the detection and tracking of birds becomes the most vital part of the implementation of the proposed system. In the subsequent section, a detailed analysis of the bird detection algorithms and the distance estimation is provided.

Figure 1: Diagrammatic representation of autonomous bird deterrence system. The black rectangular boxes highlight birds approaching the boundary (shown in red). Stereo cameras will be installed on the ground to localize the approaching birds. The black dotted line represents the field of view of the cameras. The cameras will be placed at the back of the boundary. The drone shown on the right side of the image can be controlled autonomously to patrol the boundary once the distance of the approaching birds reaches a threshold distance.

Object Detection

Bird detection is a vital part of the proposed system. The biggest problem is that birds must be detected at considerable distance. We assume that birds are the only objects that move and background subtraction techniques can be used to differentiate the background from the foreground.[1] Background subtraction is a widely used approach for detecting moving objects from static cameras (Piccardi, 2004). In general, the background image must be a representation of the scene with nothing else moving except the object which is being detected. The background subtraction technique geneally consists of two steps: (1) background model initialization and (2) model update.

In the background model initialization step, a background is initialized, which will be subtracted from the subsequent frames in order to separate the background from the foreground. In the model update step, the initialized model is updated regularly to remove the detection of slow moving objects (e.g., camera noise or periodic motion of leaves). In the Smart Scarecrow system, such removal is important and primarily assists with semi-regular background movement. The movement of leaves is also classified as moving object and therefore increases the false positives. However, over time the movement of the leaves will follow a fixed distribution and therefore (ideally) be classified as the background in the model update step. A comparison of various background subtraction algorithms was performed and tested on video feeds obtained from grape vineyards. The following subsections describe each algorithm that is compared.

Mixture of Gaussians (MOG)

This method of background subtraction is commonly known as MOG (Zivkovic, 2004; Zivkovic and Van Der Heijden, 2006). Over time, background objects appear at certain pixels in the frame. However, various changes in the background are not same and their appearance is faster than the background update. In such cases, models with a single valued background does not work. Stauffer and Grimson proposed a model in which they defined the probability of observing a certain pixel value, x at time t by means of mixture of Gaussians (1). In this each Gaussian, k, describes only one of the observable background or foreground objects.

$$P(x_t) = \sum_{i=1}^{k} \omega_{i,t} \eta \left(x_t - \mu_{i,t}, \Sigma_{i,t}\right) \tag{1}$$

To separate the background and foreground objects, all the distributions are ranked based on the assumption that the higher is the distribution, the more is the probability for it to be background. The first B distributions are ranked in order

[1] Depending on the setting, additional processing may be used to filter out other irregularly moving objects, such as planes or cars on a road.

that satisfies the equation 2, where T is the threshold and B distributions are accepted as background. Hence, this algorithm proves to be a good method to subtract background when multiple things are chammging and there is a multivariate background.

$$\sum_{i=1}^{B} \omega_i > T \qquad (2)$$

This algorithm was implemented using OpenCV libraries and code was written in python.

Figure 2: Mask created by applying the MOG algorithm on the bird video feed. Left: the actual video feed and the black dots are the birds in the frame. Right: the background subtraction mask created by the algorithm; the white pixelated dots represent the birds separated from the background.

Gaussian Mixture Gradient (GMG)

This method of background subtraction combines background image estimation and per-pixel Bayesian estimation (Godbehere et al., 2012). It uses the first few frames (usually 120) to initialize the model and adapt to it. It then applies probabilistic foreground segmentation that identifies possible foreground objects using Bayesian inference.

The algorithm involves the fixation of new samples. This fixation means that there is more weight given to new samples than to old ones. The estimates are adaptive and the newer observations are weighted more than the older ones. The improved algorithm is capable of selecting a required number of components for each pixel. In this way, it fully adapts to the scene with time. The algorithm was implemented using OpenCV libraries and code was written in Python. The image was smoothed using a morphology opening function in OpenCV, which is a combination of two filters applied sequentially on the image, erosion followed by dilation. The parameters to tune in the filter are the size of the kernel, which after observations was set to be the ellipse of size 3×3. The reason for choosing 3 as the kernel size was to remove noise without drastically removing the true positives.

Figure 3: Mask created by applying the GMG algorithm on the bird video feed. Left: the real video feed where the black dots are the birds in the frame. Right: the background subtraction mask created by GMG where the white dots represent the birds.

Visual Background Extractor (ViBe)

This algorithm is different from the previous algorithms in multiple ways. In the algorithms discussed above, the model is selected in consideration of the probability density function, where the mean and variance of each pixel in the background is calculated. However, the relevance of the Gaussian model is debatable as some authors claim that natural images exhibit non-Gaussian statistics (Shrivastava et al., 2003). In ViBe (Barnich and Van Droogenbroeck, 2011; Van Droogenbroeck and Barnich, 2014) the statistical significance of the pixel is updated, and the model is built with real pixel values rather than considering only the mean and variances of the pixels. In this approach, the classification of the new pixel value is done with respect to its immediate background pixels.

As in as the previous algorithms, the model was initialized from the first 120 frames of the video, and thus it collects a significant amount of data to separate the foreground from the background. One challenge in the Smart Scarecrow system will be to handle sudden illumination changes — foreground detection must be continual and uninterrupted. In ViBe, the initialization is done from the single frame (Barnich and Van Droogenbroeck, 2011). Thus, if the existing model is discarded and a new model is initialized, the problem of uninterrupted foreground detection can be solved and the problem of sudden illumination changes can be addressed. In terms of the model updating scheme, the algorithm performs a memoryless update of the sample which smoothly decays the samples, which are stored in the background pixel models. It considers whether neighboring background pixels have a similar temporal distribution and the new background pixel will also update the models of the neighboring pixel (Jodoin et al., 2007). Therefore, memoryless update scheme is a scheme of extracting spatial information in order to get information regarding background evolution into background pixel models masked locally by foreground (Criminisi et al., 2004). The spatial complexity of the algorithm is low since samples of frames are not stored in memory for model updates. It was implemented in C++ and the algorithm was tested on samples of videos obtained from grape vineyards. A median blur filter was used to smooth detection and remove noise in the frame. The parameter to tune the function was the aperture linear size and the value was chosen to be 3, as upon increasing the size of the kernel to greater values like 5 or 7 a lot of true positives were also removed from the frame.

Figure 4: Mask created by applying the ViBe algorithm on the bird video feed. Left: the real video feed where the black dots are the birds. Right: the background subtraction mask created by ViBe here white dots represent the birds which are separated from the background.

Split Window technique

The videos needed to detect birds are of very high resolution and therefore these algorithms can require significant computation. Since the birds are really small in size (typically less than 15×15 pixels), a very high-resolution (1920×1080 pixels) video format is used to detect the birds. Such a high resolution has a huge number of pixels to be dealt with. In order to increase the computational efficiency, we introduce a method for splitting videos in real-time. The video stream was separated into frames and background subtractor algorithm was run on each window in parallel. This was achieved using Robot Operating System (ROS). In ROS framework, each split window is treated as a separate program running simultaneously in a multithreaded architecture, automatically taking advantage of advances in modern processors. Figure 5 shows how frames are split and processed in parallel.

The videos that were used to evaluate the algorithms mentioned above were used to see the results of the split window technique. We used 30 second videos, of 1800 frames in total, which originally took 73 seconds. The same video took only 62 seconds when split into 4 frames. This test was conducted on Intel® Core™ i7 6700 HQ processor with 8 GB RAM. Further increases in speed, if required, could be achieved through using a higher-speed (although still consumer-level) chip, or by increasing the parallization of the alogrithms and leveraging graphics processing units (GPUs).

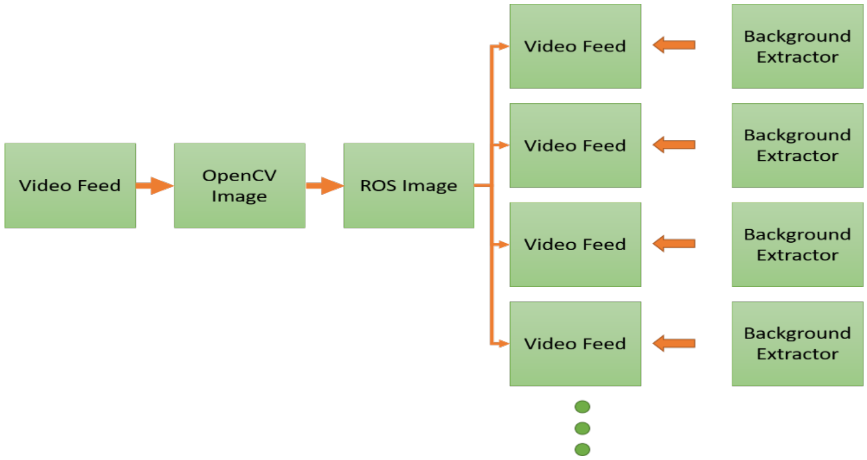

Figure 5: The figure shows the split window technique. Each part of the video can be processed separately in a multi-threaded architecture.

Distance Estimation

After detecting an object of interest (i.e., a bird), its location must be estimated. This section discusses the method used to estimate the distance of the bird from the camera (and, therefore, the relative location from a drone). As discussed in the introduction, there are a variety of methods to estimate the distance of obstacles in the real world. To make the system cost-effective, we focus on cameras. The next subsections discuss the stereo vision in general and later talk about the method used to estimate the distance of the bird using stereo vision.

Stereo Vision: Overview

The idea of using two views of the scene from the camera to extract the depth information has long been understood; Sir Charles Wheatstone invented the stereoscope in 1832 (Encyclopedia Britannica, 1987). A stereo vision system combines images from two perspectives and joins them together to get a depth view. The idea behind using stereo vision in robotics is to get depth perception.

In a stereo vision system, two cameras separated by a distance known as the baseline. The leading stereo cameras available in the market do not give the depth information further than 20 meters. Therefore, to satisfy the requirements of the Smart Scarecrow system, we must construct a custom stereo vision system. Towards this end, we test an existing depth algorithm with two low-cost webcams (Logitech C290) using a baseline of 1.4 meters. The reason for choosing such a baseline was to estimate larger distances than existing available camera setups. Because the focal length of the cameras used was quite low (3.67 mm), a larger baseline is required. A shorter baseline leads to an error in the precision of the depth at further distances. However, if the baseline becomes too large, stereo matching (Okutomi and Kanade, 1993) may become a problem.

Camera Calibration

Camera calibration is the first step towards stereo vision system. It is done by calibrating the cameras individually and then calibrating the stereo pair together. The individual camera calibration provides the numerical values for the intrinsic camera parameters: focal length, distortion ratio, and axis length. The calibration of the stereo pair provides the numerical values of the extrinsic parameters such as relative orientation and translation between the two cameras (Howard and Rogers, 1995). In principle, any characterized object can be used as a calibration target, if the 3D world coordinates of the target are known in the frame of reference of the camera (Gary and Adrian, 2008). To calibrate the camera an 8×5 checkerboard is used. The implementation was in OpenCV and code was written in python, as described in MATLAB camera calibration toolkit (Zhang, 2000). A set of 40 images is used to calibrate the cameras. The camera matrices obtained are as follows:

$$\begin{bmatrix} f_c & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \quad (2)$$

The parameters c_x and c_y represent the image center and the parameters f_x and f_y represent the horizontal and vertical scale factors. The OpenCV function *cv2.stereoCalibrate* was used to calibrate the stereo pair after obtaining the camera matrices from individual calibrations. This takes the camera matrices along with the distortion coefficient of each camera as inputs and then outputs the rotation and translational of the two cameras with respect to each other.

Depth Map Estimation

The matrices obtained after camera calibration are used for the projection of the image coordinates to the object coordinates. This is accomplished using disparity values of each pixel, because each pixel has a corresponding *(X,Y,Z)* coordinate that relates its position to the position in the *3D* space. The *Z* coordinate is found using the following equation:

$$Z = \frac{bf}{d \times sensor\ element\ size} \tag{3}$$

In equation (3), the value of the disparity is multiplied by the actual size of an individual calibration of the camera, this converts the disparity to the distance, b is the distance between two optical centers of the camera, earlier referred as baseline and f is the focal length of the camera. The *X* and *Y* values are found by using the size of the pixel at its calculated distance found for *Z*. The projection matrices obtained are the inputs to the *OpenCV* function *cv2.triangulatePoints.* This function calculates the position of the object in the real world coordinates.

The overall description of the algorithm is described in figure 6. The left and the right video feeds are fed into the algorithm. The autofocus feature of the camera is disabled to help make it robust to illumination changes. The background extractor, as discussed previously, is applied to the video feed. The largest contour in the mask obtained after the application of background extractor is the object of interest and its centroid is calculated. The centroid obtained is in pixel coordinates, which are fed into the projection matrices obtained through stereo calibration, which converts them to image coordinates in the world frame (taking the left camera as the frame of reference). The norm of the image coordinates is calculated, which is the actual distance of the moving object from the camera.

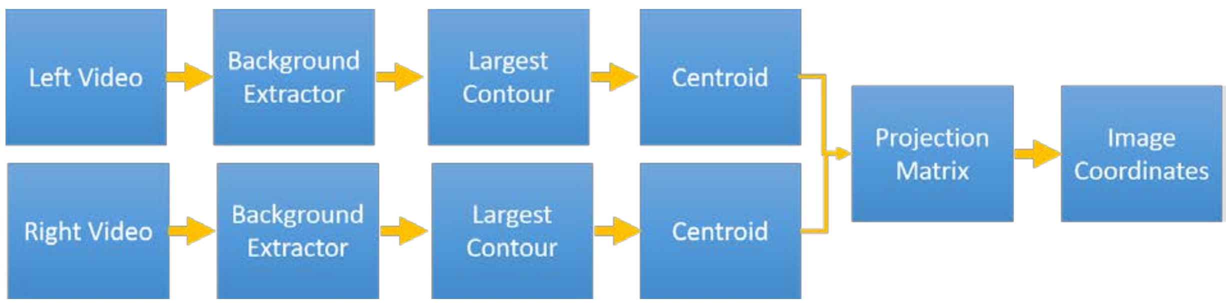

Figure 6: Overall stereo vision methodology

Results and Discussion

In order to perform analysis, a defined set of video frames was chosen to compare how each algorithm performs. Two videos of 30 seconds each were considered. In the first video, there is minimal bird activity and in the second one, there was significant bird activity as observed by human annotators. To compare the algorithm on the scale of the number of birds detected, a ground truth was established in which a particular frame for a 30 second video with significant bird activity. In particularl, the 1417th frame in the video had the highest number of birds. The frame was analyzed using all the three algorithms and the number of birds counted in each was recorded.

The resulting number of objects detected by the algorithms included noise, including some leaves and could movement. Figure 7 shows the comparisons and the dotted line shows the actual number of birds in that frame. This ground truth was calculated by averaging the results of 5 subjects manually counting the number of birds. The figure shows the number of contours detected by each algorithm in the frame. The total number of contours are the total number of objects (birds and other moving objects) that can be detected by each algorithm. Blue shows that an actual number of birds detected (true positive) and the highlight in red shows the noise detected (false positive). The comparison shows that GMG estimates the highest number of birds, but the algorithm also records a significant amount of noise (shown in red), whereas the other two algorithms perform lower than GMG in terms of the number of birds detected but have relatively very low noise. To compare the computational efficiency of the algorithms, the time of execution was also recorded, and a mean of 5 test runs was compared by running the algorithm over the entire span of the video (30 seconds). The algorithms were run on the Intel® Core™ i7 6700HQ processor, with *8* GB RAM. The results are summarized in Table 1.

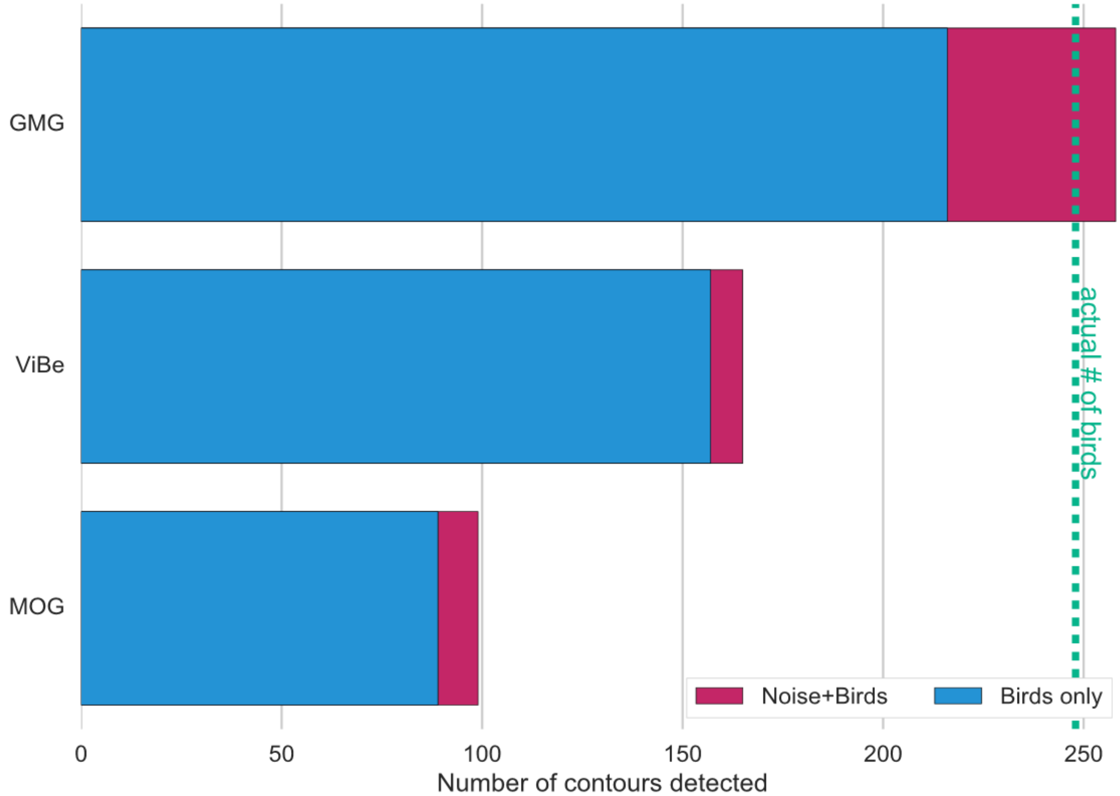

Figure 7: Graph representing the comparison of object detection algorithms. The green dotted line shows the ground truth (the actual number of birds in the frame). Blue shows the number of birds correctly detected by each algorithm (true positives) and red shows number of birds spuriously detected (false positives).

Table 1: Computational efficiency of algorithms

ALGORITHMS	# OF FRAMES	EXECUTION TIME (s)
Mixture of Gaussian (MOG)	1800	79
Gaussian Mixture Gradient (GMG)	1800	189
Visual Background Extractor (ViBe)	1800	73

The comparison shows that ViBe takes the least time to execute, whereas GMG fairs poorly in terms of computational efficiency. In the Smart Scarecrow system, real-time analysis must be performed — it would be implausible to use GMG in the design of the intelligent system, even though it is more accurate in terms of the number of birds detected.

To analyze our initial implementation of the stereo system, we tested using a small remote-controlled car. The frame rate of the video was set to 30 frames per second, which result in the frame rate of around 22-28 fps in real system implementation. The configuration of the system was stated previously. The object distance generally ran in real time, but if the car reached high speeds, the system registered lag. The distance was measured up to an accuracy of 30 centimeters at a range of 2-7 meters.

Conclusions and Future work

A comparison of the object detection methods was performed and found out that among the background subtractors compared ViBe performs the best in vineyard environment with an accuracy of 63.3%. An approach of the splitting window was proposed and evaluated to improve the computational efficiency of the background subtractor and it improved the efficiency of *12.6%*.

A machine vision system to estimate the distance of the object through stereo vision system was developed and evaluated in a simulation environment. A stereo pair with a larger baseline and with the use of cameras with better focal length would need to be used to estimate longer distances (Gallup et al., 2008), as in the case of the Smart Scarecrow system. Apart from the baseline, the focal length of the stereo camera is also a distinguishing factor in determining the accurate measure of the distance of the object of interest (Kohsiek W et al., 2002). Increasing the focal length will allow better distance estimation for much farther objects without using a huge baseline.

In the future, using the algorithm of distance estimation as stated in the paper, better quality cameras will be used to create a stereo pair to test the distance estimation of the birds. The stereo cameras will be deployed in fields to measure the accuracy and strength of the algorithm. Additional future work includes using machine learning algorithms to improve the efficiency of the bird detection algorithms and to tune our detection or distance estimation methods to better apply to bird-sized objects.

References

Anderson, A., et al. “Bird damage to select fruit crops: the cost of damage and the benefits of control in five states.” *Crop Protection* 52 (2013): 103-109.

Jarvis, Ray. “Range sensing for computer vision.” *Three-dimensional object recognition systems* 1 (1993): 17-56.

Karila, Kirsi, et al. “Feasibility of Multispectral Airborne Laser Scanning Data for Road Mapping.” *IEEE Geoscience and Remote Sensing Letters* 14.3 (2017): 294-298.

Larkin, Ronald P., and Lawrence Eisenberg. “A method for automatically detecting birds on radar.” *Bird-banding* 49.2 (1978): 172-181.

DeTech Inc., D. (2016). Bird and bat radar systems. http://www.detect-inc.com/avian.html.

Nedevschi, Sergiu, et al. “High accuracy stereo vision system for far distance obstacle detection.” *Intelligent Vehicles Symposium, 2004 IEEE*. IEEE, 2004.

Ruelle, P., and R. L. Bruggers. “Traditional approaches for protecting cereal crops from birds in Africa.” (1982).

Marsh, Rex E., William A. Erickson, and Terrell P. Salmon. “Scarecrows and predator models for frightening birds from specific areas.” (1992).

Bishop, J., et al. “Review of international research literature regarding the effectiveness of auditory bird scaring techniques and potential alternatives.” *Food and Rural Affairs, York, UK* (2003).

Tripicchio, Paolo, et al. “Towards smart farming and sustainable agriculture with drones.” *Intelligent Environments (IE), 2015 International Conference on*. IEEE, 2015.

Davidson, Malcolm WJ, et al. “On the characterization of agricultural soil roughness for radar remote sensing studies.” *IEEE Transactions on Geoscience and Remote Sensing* 38.2 (2000): 630-640.

Bac, C. Wouter, et al. “Harvesting Robots for High-value Crops: State of the art Review and Challenges Ahead.” *Journal of Field Robotics* 31.6 (2014): 888-911.

Ampatzidis, Yiannis, Joshua Ward, and Omar Samara. “Autonomous System for Pest Bird Control in Specialty Crops using Unmanned Aerial Vehicles.” *2015 ASABE Annual International Meeting*. American Society of Agricultural and Biological Engineers, 2015.

Garage, Willow. “Robot operating system (ROS).” (2012).

Anderson, Karen, and Kevin J. Gaston. “Lightweight unmanned aerial vehicles will revolutionize spatial ecology.” *Frontiers in Ecology and the Environment* 11.3 (2013): 138-146.

Vanhemert, K. (2014). Realistic robo-hawks designed to fly around and terrorize real birds. http://www.wired.com/2014/08/ realistic-robo-hawks-designed-to-fly-around-and-terrorize-real-birds/

Piccardi, Massimo. “Background subtraction techniques: a review.” *Systems, man and cybernetics, 2004 IEEE international conference on*. Vol. 4. IEEE, 2004.

Zivkovic, Zoran. “Improved adaptive Gaussian mixture model for background subtraction.” *Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on*. Vol. 2. IEEE, 2004.

Zivkovic, Zoran, and Ferdinand Van Der Heijden. “Efficient adaptive density estimation per image pixel for the task of background subtraction.” *Pattern recognition letters* 27.7 (2006): 773-780.

Godbehere, Andrew B., Akihiro Matsukawa, and Ken Goldberg. “Visual tracking of human visitors under variable-lighting conditions for a responsive audio art installation.” *American Control Conference (ACC), 2012*. IEEE, 2012.

Srivastava, Anuj, et al. “On advances in statistical modeling of natural images.” *Journal of mathematical imaging and vision* 18.1 (2003): 17-33.

Barnich, Olivier, and Marc Van Droogenbroeck. “ViBe: A universal background subtraction algorithm for video sequences.” *IEEE Transactions on Image processing* 20.6 (2011): 1709-1724.

Van Droogenbroeck, Marc, and Olivier Barnich. “ViBe: A disruptive method for background subtraction.” *Background Modeling and Foreground Detection for Video Surveillance* (2014): 7-1.

Jodoin, Pierre-Marc, Max Mignotte, and Janusz Konrad. “Statistical background subtraction using spatial cues.” *IEEE Transactions on Circuits and Systems for Video Technology* 17.12 (2007): 1758-1763.

Criminisi, Antonio, Patrick Pérez, and Kentaro Toyama. “Region filling and object removal by exemplar-based image inpainting.” *IEEE Transactions on image processing* 13.9 (2004): 1200-1212.

Howard, Ian P., and Brian J. Rogers. *Binocular vision and stereopsis*. Oxford University Press, USA, 1995.

Okutomi, Masatoshi, and Takeo Kanade. “A multiple-baseline stereo.” *IEEE Transactions on pattern analysis and machine intelligence* 15.4 (1993): 353-363.

Zhang, Zhengyou. “A flexible new technique for camera calibration.” *IEEE Transactions on pattern analysis and machine intelligence* 22.11 (2000): 1330-1334.

Pollefeys, Marc, et al. “Detailed real-time urban 3d reconstruction from video.” *International Journal of Computer Vision* 78.2 (2008): 143-167.

Kohsiek, W., et al. “An extra large aperture scintillometer for long range applications.” *Boundary-Layer Meteorology* 105.1 (2002): 119-127.