Automatic Ship Detection on Inland Waters: Problems and a Preliminary Solution

Tomasz Hyla
Marine Technology Ltd.
Szczecin, Poland
e-mail: t.hyla@marinetechnology.pl

Natalia Wawrzyniak
Institute of Geoinformatics, Faculty of Navigation
Maritime University of Szczecin
Szczecin, Poland
e-mail: n.wawrzyniak@am.szczecin.pl

***Abstract*—Marine ship traffic is usually monitored using a combination of Automatic Identification System (AIS) and radar networks. Video monitoring is often used to control vessel traffic on rivers, in ports, and other restricted areas. Despite the many means used, the task of automatic detection and identification of vessels on inland waters is not trivial. In this paper, the problem of automatic ship detection using the video surveillance system is analyzed and discussed primarily in the context of systems' performance. In the proposed solution, we assume that detection uses only video streams from the existing monitoring system without any additional hardware or special configuration or placement of cameras. In addition, the detection must work in real time and the system must detect all moving vessels, including small boats and kayaks. The main contributions of this work consist in presenting the results from several performance tests using different background subtraction algorithms as well as discussing problems that make moving vessels difficult to detect.**

Keywords- ship; detection; video surveillance; fixed camera.

I. INTRODUCTION

Ship traffic is usually monitored using a combination of radar networks and AISs. Moreover, video monitoring systems are common to monitor the traffic on inland or coastal waters. From a technical perspective, video monitoring is a passive system in contrast to a radar and AIS, which can be treated as active sensors. Therefore, traffic monitoring using a video monitoring system can be seen as a more cost efficient system that does not require additional hardware and additionally is more reliable in recognizing, e.g., terror threats, than a radar or AIS. On the other hand, automatic vessel detection is a complex task that requires to consider many scenarios.

The ship in a video stream can be detected using two basic approaches. The first one is to use a pixel-based detection method that allows detecting any moving object on constant or slightly changing background. The second approach is to use object-based detection using some kind of classifier. The second approach is better when it is possible to find a distinctive property of a class of objects, e.g., mast of a sailing vessel, because it usually provides better detection result.

One of the possible solutions to the ship detection problem was presented by Ferreira et al. [1]. The authors use two cameras: one camera with low resolution that detects movement and another camera with high resolution that is used to take a photo when the first camera detects movement. Their solution is designed to detect fishing vessels. They achieved the best results when using object-based detection based on Histogram of Oriented Gradients (HOG) classifier [2]. In contrast, Hu et al. [3] used pixel based detection in their visual surveillance scheme for cage aquaculture that automatically detects and tracks ships. They used the median scheme to create a background image from previous *N* frames with some additional improvements that allowed to reduce the influence of sea waves. The problem of ship detection in the presence of waves was also addressed by Szpak and Tapamo [4]. They present techniques that solve a problem of moving vessels' tracking in the presence of a moving dynamic background (the ocean). Other works related to the problem of ship detection include [5][6] and a survey [7].

This short paper is a part of an ongoing research in Ship Recognition (SHREC) [8], which concerns automatic recognition and identification of non-conventional (according to International Convention for the Safety of Life at Sea (SOLAS)) ships in areas covered by RIS (River Information System) and Vessel Traffic Service (VTS) systems. In this paper, we analyze the problem of ship detection on inland and coastal waters, provide a preliminary solution and test its performance. In the proposed approach, we assume that detection uses only video streams from an existing monitoring system without any additional hardware or special configuration of cameras. In addition, the detection is performed in real time with the use of only one processor working at not more than 25% of its maximum load for one video stream. This requirement is caused by the need to perform also other operations for the video stream from one camera, i.e., ship classification and identification. The system must detect all moving ships, including small boats and kayaks, which is the main difference from existing solutions that mostly focus on only one vessel type. The main contributions of this work consist in presenting the results from several performance tests using different background subtraction algorithms as well as discussing problems that make moving vessels difficult to detect.

The rest of this paper is organized as follows. Section 2 describes the most popular background subtraction algorithms. Section 3 presents the problem of ship detection on inland waters using existing video monitoring. Section 4 presents the results of performance tests of different

 ISBN: 978-1-61208-696-5

background subtraction algorithms. We conclude the paper in Section 5.

II. BACKGROUND SUBTRACTION ALGORITHMS

Moving objects can be detected using a background subtraction algorithm or, to be more precise, using a background/foreground segmentation algorithm. The task of selecting foreground objects in a scene is easy in indoor environments, but in outdoor environments it is more difficult because of many factors that must be considered. The easiest approach is to save a reference (first) frame and then calculate the difference to this frame (algorithm FF). More advanced algorithms use around 100 to 200 frames from which they try to model a background. Many such algorithms exist and it is very difficult to show which one is the best, because their accuracy depends on a chosen benchmark. Additionally, better performance might require more processing power or memory. Storing 200 decompressed frames of a full high definition video requires around 1.2GB of memory and 4K video requires around 4.8GB. The background subtraction algorithms were evaluated by [9] and compared by [10].

Several background subtraction algorithms are implemented in the OpenCV library [11]. To begin with, Gaussian Mixture-based Background- Foreground Segmentation (MOG) algorithm (that uses a mixture of *K* (*K*=3 to 5) gaussian distributions to model each background picture. The probable values of background pixels are the ones that are more static and present in more previous frames [12]. Next, Gaussian Mixture-based Background-Foreground Segmentation Algorithm version 2 (MOG2) [13] is available, which is an improved version of MOG. The algorithm selects the appropriate number of gaussian distributions for each pixel. It works better in scenes that change often, e.g., due to illumination changes caused by clouds. Shadows can also be detected using this algorithm.

The algorithm Godbehere-Matsukawa-Goldberg (GMG) [17] uses by default 120 frames for background modelling and per-pixel Bayesian segmentation. New frames have more weight to support variable lightning conditions. The unwanted noise is removed using morphological operations like closing and opening. Another algorithm, 'CouNT (CNT) was designed by Sagi Zeevi [14] to reflect the human vision. It is designed for variable outdoor lighting conditions and it works well on Internet of Things (IoT) hardware. Other algorithms include: *k* Nearest Neighbours (KNN) that implements *K*-nearest neighbors background subtraction from [15], the algorithm created during Google Summer of Code (GSOC) [11], and Background Subtraction using Local SVD Binary Pattern (LSBP) [16].

III. SHIP DETECTION

Detection of a foreground object is an easy task when background and lightning are constant at the scene and the camera is fixed. However, in our system, there are small background changes and the lighting changes over time. The basic assumption of our system is that the system uses fixed cameras, which are part of an existing surveillance system and therefore must use video streams from different camera' views.

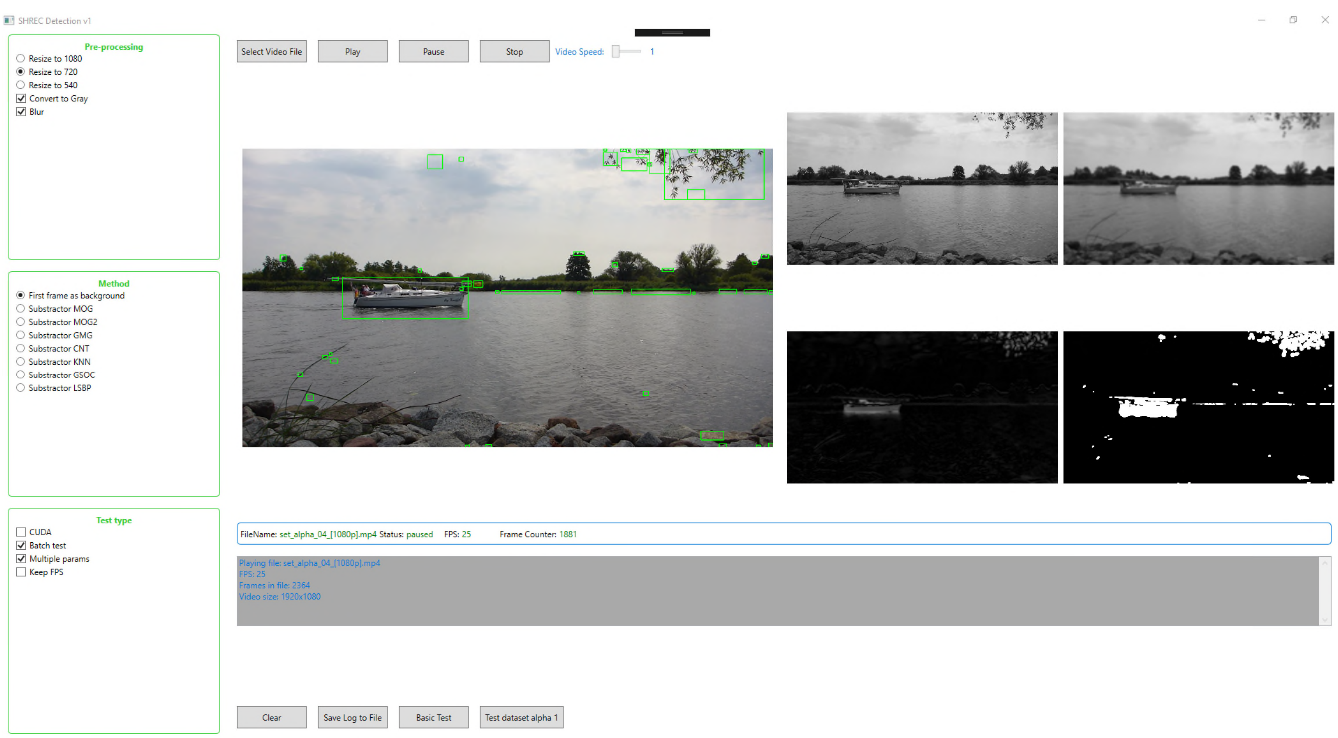

Figure 1. A screenshot from the application used to test different algorithms of background subtraction.

 ISBN: 978-1-61208-696-5

The main problems that were identified during observations of different video streams are as follows:
1) the camera is usually not directed only at the water and several other moving objects are present in the frame:
    a) trees, other plants and movement depend on wind speed and direction;
    b) object on land: cars, bikes, people, and even trains;
2) animals: especially birds, but also dogs and jumping fish;
3) different water state (waves) – however, less than at sea and waves after fast moving boats;
4) different camera angles, waterway crossroads;
5) obstacles blocking the view, e.g., pillars of bridges, street lamps, mast of moored ships, threes, buildings;
6) multiples ships coming in different directions simultaneously and overlapping with each other;
7) different lightning during different time of the day and of the year.

Our preliminary method for ships detection consists of three main steps as in most standard solutions used for object detection. The first step is the pre-processing in which video is decoded to frames in a bitmap format. The frame rate is reduced to 25 or 30 frames when a video file contains 50 or 60 frames per second. Then, the frames are resized to 1280x720 or 940x540 resolution and converted to grayscale. Additionally, all 25 or 30 frames are used as an input for the background subtraction algorithm.

The second step is foreground extraction that results in a mask. The mask is a black-white bitmap where white color means a foreground object. In this step, one of the background subtraction algorithms is used. Depending on the algorithm, an input frame might be blurred and the morphological opening and closing might be used on the output mask.

The third step is the ship distinction from all moving objects. This step is the most difficult as all artefacts must be removed. The step contains edge detection, creation of bounding boxes, removing objects from pre-configured non-water regions (e.g., regions that contains trees), removing objects with small area or dimensions, and removing objects that appeared for the first time (using a few seconds history).

IV. Experimental Results

The first step in creating the ship detection method was to test how different background subtraction algorithms with different parameters will behave on different video samples. The first decision was making pixel-based detection obligatory due to the fact that the system must detect all types of vessels including leisure units. The tests have been carried out using a test application (Figure 1) that allows testing video samples using different foreground extraction algorithms.

Figure 1 shows detection using the simplest method for foreground extraction, i.e., blurring image and calculating difference to the reference frame containing only background. It contains detected ships with some artefacts that can be easily removed in further steps.

One of the main problems related to real time detection is the performance that limits possible options, especially when the input stream is the high quality 4K stream. Therefore, the first test was a performance test. The test application is written in C# and is using Emgu CV version 3.4.3 (C# wrapper for OpenCV). Two test computers (A - Intel Core i5-8250U, 32GB RAM, SSD 512GB; B - Intel Core i7-8700K, 32GB RAM, SSD 1TB, NVIDIA Quadro P4000) were used in the test. In the test, the explicit Compute Unified Device Architecture (CUDA) OpenCV functions were not used, so it was possible to test only the impact of the algorithm on Central Processing Unit (CPU). Three samples from our database were chosen to the test:
1) Sample 1, low quality High Definition (HD) stream from webcam (1280x720, 18 fps, bitrate: 596kb/s, Advanced Video Coding (AVC) Main@L3.1, duration 90s);
2) Sample 2, medium quality Full High Definition (FHD) (1920x1080, 25 fps, bitrate: 20 Mb/s, AVC Baseline@L4, duration: 90s);
3) Sample 3, high quality 4K (3840x2160, 30fps, bitrate: 48Mb/s, AVC High@L5.1, duration 90s).

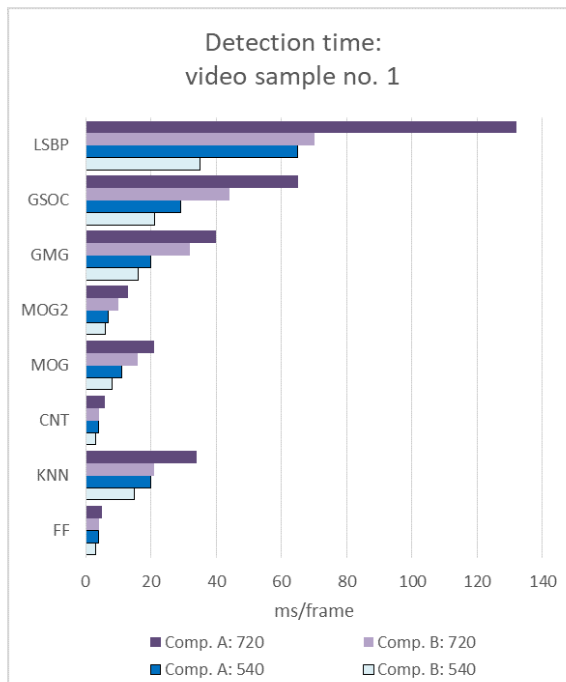

Figure 2. Detection performance - video sample no. 1.

In all the tests, samples were kept in the original size (except 4k resolution, as initial tests have shown, uses to much CPU and does not give better detection results, than stream resized to FHD resolution) or resized to resolutions 1920x1080, 1280x720, and 960x540. The test includes seven different background subtraction algorithms from OpenCV (MOG [12], MOG2 [13], GMG [17], CNT [14], KNN [15], GSOC[11], LSBP [16]) and a simple algorithm that calculates difference to frame with background only (FF).

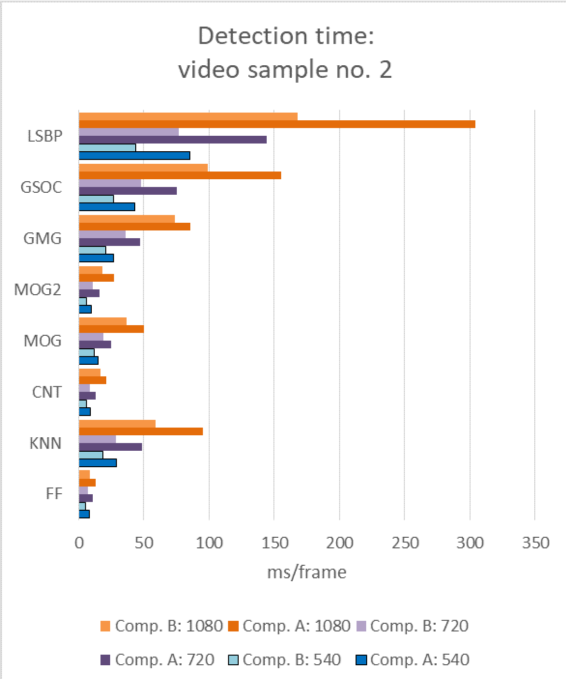

Figure 3. Detection performance - video sample no. 2.

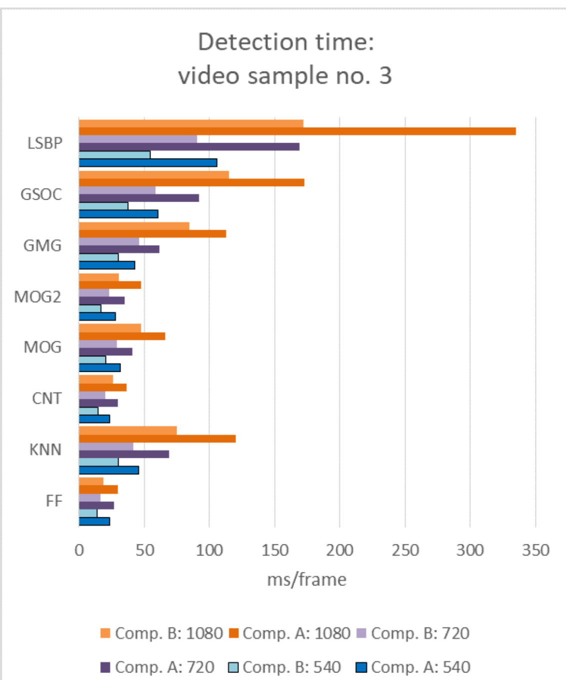

Figure 4. Detection performance - video sample no. 3.

The test results (see Figures 2-4) confirm that detection using 1920x1080 resolution takes significantly more time than using 1280x720 and 960x540 resolutions when using a more advanced algorithm. The maximum detection time must be below 40 ms to enable real time processing using the 25 fps video. In all three video samples, LSBP substractor was the slowest, where processing of 1080p frame took between 150-300 ms (depending on the test computer and input resolution). The fastest was MOG2 substractor and the simple algorithm (but ineffective in dynamic scenes) that subtracts a current frame from a first frame that contains only background (FF) with frame processing time between 3 and 30 ms. The tests have shown that the size of the input has a little impact on processing time. It is mainly, because decoding H.264 stream is fast, for the reason that AVC codec internally uses hardware acceleration and a frame resizing operation is rather simple and therefore swift.

Additionally, a second experiment was carried out to check how background subtraction algorithms affect the ship detection result. During the test, different video samples were viewed by our research team members. This was an initial experiment that allowed us to narrow down algorithms for further quality tests. The main conclusion from the experiment is that higher resolution does not always provide better detection results, mainly because higher resolution also mean more details and noise. The preliminary observation from the test is that GSOC algorithm returns the smallest number of erroneous artefacts.

V. Conclusion and Future Work

In the SHREC system, a detection phase is one of the steps that must be calculated in real time by a single processor for one camera. The test shows that a time less than 100 ms per frame can be achieved using most of the algorithms when resizing to 720p or 540p resolution is used. This time is acceptable, because no more than 3 frames (probably 1) per second will be used for detection purposes.

The main problem with background subtraction algorithms is that often, when on a scene problem described in Section 2 emerges, they do not correctly recognize a foreground object (a ship). Most of the artefacts (incorrectly recognized objects) can be easily removed, but when an object (a ship) is incorrectly subtracted from a background, it is difficult to correct it in further steps. For example, sometimes the presence of waves causes that a background subtraction does not correctly detect a slowly moving ship. The first tests on quality of detection indicate that the best algorithms are GSOC and CNT.

Future works include improving the proposed detection method. The works will be carried out in two paths. In the first path, the algorithm that returns bounding box containing detected ships will be improved based on experimental results. In second one, the method will be further optimized by shifting most of the computation to a graphic processing unit.

The method will be tested on two larger data sets that contain more than 200 video samples each. The first dataset was recorded for the purpose of the project on the waterways near Szczecin. The second contains video files recorded from several public webcams that shows different waterways in Europe. After the test, if results are not satisfactory, the algorithm that uses two background subtraction algorithms at once or one of optical flow methods will be used. The optical flow methods were not tested on the beginning due to their high performance requirements for FHD or 4K video files [18].

ACKNOWLEDGEMENT

This scientific research work was supported by National Centre for Research and Development (NCBR) of Poland under grant No. LIDER/17/0098/L-8/16/NCBR/2017) and under grant No. 1/S/IG/16 financed from a subsidy of the Ministry of Science and Higher Education for statutory activities.

REFERENCES

[1] J. C. Ferreira, J. Branquinho, P. C. Ferreira, and F. Piedade, “Computer vision algorithms fishing vessel monitoring—identification of vesselplate number,” Ambient Intelligence– Software and Applications – 8th International Symposium on Ambient Intelligence (ISAmI 2017), J. F. De Paz, V. Juli´an, G. Villarrubia, G. Marreiros, and P. Novais, Eds. Cham: Springer International Publishing, 2017, pp. 9–17, 10.1007/978-3-319-61118-1_2.

[2] R. K. McConnell, “Method of and apparatus for pattern recognition”, US Patent 4,567,610, 1986.

[3] W.-C. Hu, C.-Y. Yang, and D.-Y. Huang, “Robust real-time ship detection and tracking for visual surveillance of cage aquaculture,” Journal of Visual Communication and Image Representation, vol. 22, no. 6, 2011, pp. 543–556, doi:10.1016/j.jvcir.2011.03.009.

[4] Z. L. Szpak and J. R. Tapamo, “Maritime surveillance: Tracking ships inside a dynamic background using a fast level-set,” Expert Systems with Applications, vol. 38, no. 6, 2011, pp. 6669–6680, doi: 10.1016/j.eswa.2010.11.068.

[5] N. Kaido, S. Yamamoto, and T. Hashimoto, “Examination of automatic detection and tracking of ships on camera image in marine environment,” 2016 Techno-Ocean (Techno-Ocean), Oct. 2016, pp. 58–63, doi:10.1109/Techno-Ocean.2016.7890748.

[6] Y. J. Kim, Y. K. Chung, and B. G. Lee, “Vessel tracking vision system using a combination of kaiman filter, bayesian classification, and adaptive tracking algorithm,” 16th International Conference on Advanced Communication Technology, Feb. 2014, pp. 196–201, doi:10.1109/ICACT.2014.6778948.

[7] R. da Silva Moreira, N. F. F. Ebecken, A. S. Alves, F. Livernet, and A. Campillo-Navetti, “A survey on video detection and tracking of maritime vessels,” International Journal of Research and Reviews in Applied Sciences, vol. 20, no. 1, July 2014, pp. 37–50.

[8] N.Wawrzyniak and A. Stateczny, “Automatic watercraft recognition and identification on water areas covered by video monitoring as extension for sea and river traffic supervision systems,” Polish Maritime Research, vol. 25, iss. s1, June 2018, pp. 5-13, doi.org/10.2478/pomr-2018-0016.

[9] S. Brutzer, B. Höferlin and G. Heidemann, "Evaluation of background subtraction techniques for video surveillance," CVPR 2011, Colorado Springs, CO, USA, 2011, pp. 1937-1944, doi: 10.1109/CVPR.2011.5995508.

[10] Y. Benezeth, Y. Benezeth, Pierre-Marc Jodoin, Pierre-Marc Jodoin, Bruno Emile, Bruno Emile, Helene Laurent, Helene Laurent, Christophe Rosenberger, Christophe Rosenberger, "Comparative study of background subtraction algorithms," Journal of Electronic Imaging vol. 19, no. 3, July 2010, pp. 1-30, https://doi.org/10.1117/1.3456695.

[11] “Emgu CV Library Documentation version 3.4.3,” http://www.emgu.com/wiki/files/3.4.3/document/index.html, 2018, [retrieved: February, 2019].

[12] P. KaewTraKulPong and R. Bowden, “An Improved Adaptive Background Mixture Model for Real-time Tracking with Shadow Detection,” Boston, MA: Springer US, 2002, pp. 135–144, doi: 10.1007/978-1-4615-0913-4_11.

[13] Z. Zivkovic, “Improved adaptive gaussian mixture model for background subtraction,” in Proceedings of the Pattern Recognition, 17th International Conference on (ICPR’04) Volume 2 - Volume 02, ser. ICPR ’04. Washington, DC, USA: IEEE Computer Society, 2004, pp. 28–31, doi:10.1109/ICPR.2004.479.

[14] S. Zeevi, BackgroundSubtractorCNT Project, https://sagi-.github.io/BackgroundSubtractorCNT/, [retrieved: February, 2019].

[15] Z. Zivkovic and F. van der Heijden, “Efficient adaptive density estimation per image pixel for the task of background subtraction,” Pattern Recognition Letters, vol. 27, no. 7, 2006, pp. 773–780.

[16] L. Guo, D. Xu and Z. Qiang, "Background Subtraction Using Local SVD Binary Pattern," 2016 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Las Vegas, NV, 2016, pp. 1159-1167, doi: 10.1109/CVPRW.2016.148.

[17] A. B. Godbehere, A. Matsukawa, and K. Goldberg, “Visual tracking of human visitors under variable-lighting conditions for a responsive audio art installation,” in 2012 American Control Conference (ACC), June 2012, pp. 4305–4312, doi:10.1109/ACC.2012.6315174.

[18] J. Huang, W. Zou, J. Zhu, and Z Zhu, “Optical Flow Based Real-time Moving Object Detection in Unconstrained Scenes,” arXiv e-prints, July 2018, http://adsabs.harvard.edu/abs/2018arXiv180704890H, [retrieved: February, 2019].

