Betancourt A., Morerio P., Marcenaro L., Rauterberg M., Regazzoni CS. (2015). Filtering SVM frame-by-frame binary classification in a detection framework. In: Proceedings of International Conference on Image Processing - IEEE ICIP (pp. 2552-2556). Quebec, Canada: IEEE Press.

FILTERING SVM FRAME-BY-FRAME BINARY CLASSIFICATION IN A DETECTION FRAMEWORK

Alejandro Betancourt[1,2], *Pietro Morerio*[1], *Lucio Marcenaro*[1], *Matthias Rauterberg*[2], *Carlo Regazzoni*[1]

[1]Information and Signal Processing for Cognitive Telecommunications Group.
Department of Naval, Electric, Electronic and Telecommunications Engineering.
University of Genoa, Italy

[2] Designed Intelligence Group.
Department of Industrial Design.
Eindhoven University of Technology.
Eindhoven, Netherlands.

ABSTRACT

Classifying frames, or parts of them, is a common way of carrying out detection tasks in computer vision. However, frame by frame classification suffers from sudden significant variations in image texture, colour and luminosity, resulting in noise in the extracted features and consequently in the decisions taken. Support Vector Machines have been widely validated as powerful tools for frame by frame detection of non-separable datasets, but are extremely sensitive to these variations between adjacent frames, creating as consequence sudden flickering in the classification results. This work proposes a Dynamic Bayesian Network to smooth the classification results of Support Vector Machines (SVM) in detection tasks. The method is evaluated in First Person Vision (FPV) videos, where a SVM is used to decide whether or not the user's hands are in his field of view.

Index Terms— Classification; Detection; Bayesian Filtering; Hand detection; First Person Vision; Egocentric Vision; Wearable computing

1. INTRODUCTION

Classification-for-detection is a widely studied area in computer science. Its main objective is to decide whether a particular object O is present in the environment. The variety of objects to detect is broad and multiple applications were investigated, such as pedestrian detection [1, 2, 3, 4, 5], hand detection [6], face detection [7], intrusion detection [8], among others. In computer vision, a common approach to detect O in an image (or in a video sequence) is to exploit a classifier under a supervised framework, using a balanced training dataset with O and non-O sample images. In particular, samples (especially non-O) should be sufficiently heterogeneous in order to allow a good discrimination of the two classes.

The problem of *detection* is often related to the *localization* of an object in a frame (equivalently, the problem can be formulated as the detection of an object in a localized sub-part of the frame). This task is frequently faced in an iterative way, classifying images framed by a sliding window of different sizes moving across the image. All these approaches are derived from the seminal work by Viola and Jones [9], in turn inspired by [10]. Despite being computationally expensive, these strategies are widely accepted as a powerful strategy for object detection and localization.

This work was partially supported by the Erasmus Mundus joint Doctorate in Interactive and Cognitive Environments, which is funded by the EACEA, Agency of the European Commission under EMJD ICE.

Instead of classifying raw images directly, it is usually preferable to classify extracted features. Multiple alternatives have been previously evaluated depending of the detection goal. An extensive literature is available: some of the more popular image features are color histograms [11, 12] to detect parts of the human body, global features as GIST [13] to detect general properties of the scene, rotation and scale invariant features as SIFT [14] to detect and identify multiple objects at different scales and positions, and shape features as Histogram Oriented Gradients (HOG) [2] to exploit particular characteristic in the shape of objects. Recent approaches use mixtures of features at different levels under the deep learning framework [15]. Regarding the classifiers, multiple alternatives are available. However, a general consensus has been achieved about the powerful combination between HOG and Support Vector Machines (SVM), particularly for non separable datasets [2, 6].

These approaches are developed and trained without using temporal information, therefore their application in video sequences is usually carried out as a naive frame by frame classification [16], which is extremely sensitive to small frame-to-frame features' variations. To alleviate this problem some researchers smooth the features to reduce their spatial and temporal variations [17]. Temporal stability of the detections can be considered as a common goal for many video processing applications, thus a dynamic smoothing is typically more important than the spatial approach. For instance, this is done for depth videos [18] and for RGB first-person videos [19] at pixel level.

This paper presents a Dynamic Bayesian Network (DBN) to smooth the classification-decision process within the detection problem. The proposed method relies on the theoretical construction of the SVM to exploit the level of uncertainty of the decision. The approach is computationally effective, because, by using the classification certainty of the SVM, it avoids filtering a multidimensional vector of features. Namely, we propose to move the filtering step at a higher hierarchical level in the estimation process (Figure 1). In fact, we point out that different hierarchical levels of features can be extracted from an image, starting from trivial pixel level ones, to end up with global color histograms, GIST, HOG, among others. Eventually, the output of a classifier can be accounted for as a high level feature. In addition, our approach is almost independent of the underlying feature level, meaning that it could easily be adapted to new features. The proposed method is suitable for the general problem of object detection, however its applicability is particularly effective for highly variable signals like FPV videos in which noise is high. See [20] for a general overview of FPV methods. To illustrate the performance of the DBN we extend the frame by frame hand-detector proposed by [6] and compare the results. In the evaluation procedure

we use, as the original work suggest, HOG, GIST and multiple color spaces. However, due to theoretical issues, only SVM is used for classification. The authors in [21] briefly summarize the importance of a reliable hand-detector to develop hand-based methods in FPV.

The remainder of this paper is organized as follows: Section 2 presents the DBN network and each of its parts, summarizing in section 2.1 and 2.2 the basic concepts behind SVM and Kalman filter respectively, and explaining how to use them in the detection problem. Finally, section 3 extends the work of [21] using the proposed DBN. Section 4 concludes and highlights some research lines.

2. FILTERING THE DECISION PROCESS

In this section, a SVM-based detector is extended with dynamic information using the DBN proposed in Figure 1, which sketches a multiple-level Bayesian filter. Here, we assume the measurement $z_k \in \mathbb{R}$ as the result of applying the SVM to set of features F_k extracted from the k^{th} frame I_k (this is detailed in Section 2.1). The state $x_k \in R^2$ includes the signed distance from the decision boundary of the SVM, enriched with its speed: $x_k = [f(F_k), \dot{f}(F_k)]$. In the upper level lies the binary variable h_k taking the value 1 for detection and -1 otherwise. The dotted line is drawn to illustrate the possible filtering at features level, as discussed in the previous section. However, in our case only the upper level is filtered.

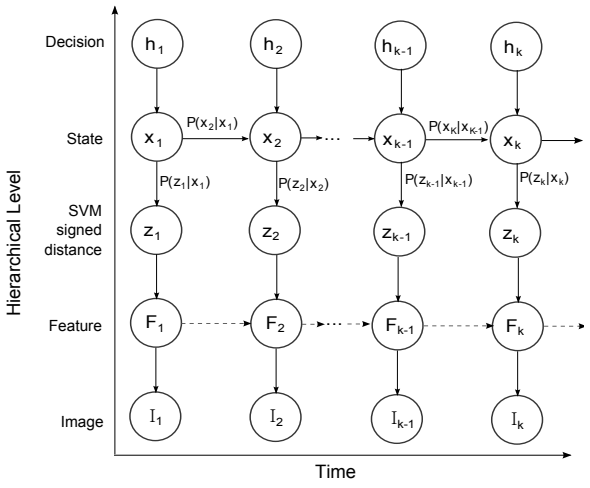


Fig. 1: Dynamic Bayesian Network for smoothing the decision process.

To quickly illustrate the dynamic filtering procedure, let us assume the knowledge of the state x_{k-1}. Then using the process model $P(x_k|x_{k-1})$ we predict $\hat{x}_k^-$ (*a priori* estimate). Once the measurement z_k is available, we update the estimate $\hat{x}_k$ using the measurement function $P(z_k|x_k)$. Subsequently the presence (or not) of hands in the frame k is decided, using $sign(\hat{x}_k[0])$ (taking the $sign$ is equivalent to a decision threshold equal to 0).

The measurement is the real valued output of the classifier as explained in Section 2.1, which briefly introduces the SVM as a tool to classify non separable data, and explains the meaning of the its coefficients and its relation with the detection problem. For more details about the SVM the reader can refer to [22]. The process and measurement models are defined by a linear Kalman filter with constant acceleration. Section 2.2 introduces the discrete Kalman filter and its iterative equations. For more details about its probabilistic formulation please refer to [23].

2.1. Support Vector Machine

Let's assume a dataset composed by training data of N pairs $(F_1, y_1), (F_2, y_2), \ldots, (F_N, y_N)$, with $F_i \in R^p$ and $y_i \in \{-1, 1\}$. Equation (1) defines a classification hyperplane and equation (2) its induced classification rule, where β is a unit vector.

$$\{F : f(F) = F^T\beta + \beta_0 = 0\} \tag{1}$$

$$G(F) = sign(f(F)) = sign(F^T\beta + \beta_0) \tag{2}$$

If the classes are separable then $f(F)$ is the signed distance from a point F to the separation hyperplane, and the solution of the optimization problem given by (3) maximizes the margin between the training points.

$$\min_{\beta,\beta_0} ||\beta|| \text{ subject to: } y_i(F_i^T\beta + \beta_0) \geq 1, \forall i \tag{3}$$

$$\min_{\beta,\beta_0} ||\beta|| \text{ subject to: } y_i(F_i^T\beta + \beta_0) \geq 1 - \xi_i, \forall i, \quad \xi_i \geq 0, \sum \xi_i \leq K \tag{4}$$

If the classes are not separable, it is possible to allow some points to be misclassified and reformulate (3) as (4), where $\xi = (\xi_1, \xi_2, \ldots, \xi_N)$ are referred to as the slack variables, and K is constant. Note that ξ_i is defined by all the points in the training set $\forall i$; however, it only takes values different from zero for those points that fall in the decision boundary or are misclassified. These points are the support vectors.

Returning to the *detection* problem, the solution of (4) gives us a set of coefficients which can be used in (2) as decision criteria. However, given the final objective this paper, we use the signed distance $f(F_k)$ to the classification hyperplane as the measurement z_k, where F_k is a global feature extracted from the k-th frame (F_k in the proposed DBN). It is important to note that the signed distance to the decision boundary $f(F)$ gives both a description of the result $G(F)$ of the classification (i.e $sign(f(F))$) as well as its level of certainty: in simple words, the larger the distance the more confident the decision (i.e. the less its covariance). Keeping the two pieces of information together allows to have a continuous state variable to be filtered, instead of a discrete one ($G(F)$ is indeed binary). In addition, augmenting the state with $\dot{f}(F)$ allows to control sudden variations of such confidence. In some sense the filter is thus self-aware of how good the classification is evolving, which can introduce some feedback mechanism to compensate for poor classification.

2.2. Kalman filter

The previous section proposed how to model the classification state by including its confidence. This section explains how to transfer and stabilize the state from time to time, to reduce the number of wrong decisions caused by little variations in the features between frames. For this purpose we use a discrete linear Kalman filter. In general notation, the process model is given by (5) and the measurement model by (6).

$$x_k = Ax_{k-1} + w_{k-1}, \tag{5}$$
$$z_k = Hx_k + v_k, \tag{6}$$

where $x_k \in \mathbb{R}^n$ is the state and $z_k \in \mathbb{R}^m$ is the measurement. The matrix $A_{n\times n}$ relates the state at previous step $k-1$ to the state at current step k. The matrix $H_{m\times n}$ relates the state with the measurement. The random variables w and v are the process and measurement noise respectively, which are assumed Gaussian with zero mean and covariances equal to $Q_{n\times n}$ and $R_{m\times m}$ respectively.

Based on these equations, the prediction stage is given by (7), which, using the current values of $\hat{x}_{k-1}$ and P_{k-1} approximates their next values $\hat{x}_k^-$ and P_k^-. P_k is the error covariance at time k and $\hat{x}$ is an estimator of x.

$$\begin{aligned} \hat{x}_k^- &= A\hat{x}_{k-1} \\ P_k^- &= AP_{k-1}A^T + Q \end{aligned} \tag{7}$$

Once a new measurement is available the values of x_k and P_k are updated using (8), where K is known as the Kalman gain.

$$\begin{aligned} K_k &= P_k^- H^T (HP_k^- H^T + R)^{-1} \\ \hat{x}_k &= \hat{x}_k^- + K_k(z_k - H\hat{x}_k^-) \\ P_k &= (I - K_k H)P_k^- \end{aligned} \tag{8}$$

At this point it is possible to use $\hat{x}_k$ and P_k for a new predicting stage. In the DBN, $\hat{x}_k$ is a two dimensional vector, and is used to decide the value of h_k by taking $\hat{h}_k = sign(\hat{x}_k[0])$ (as already mentioned, this is equivalent to have a decision threshold equal to 0).

Ultimately, extracted features which are really close to the decision boundary can jump from one side to the other in consecutive frames, being their (signed) measured distance z_k slightly positive or slightly negative. Filtering such a distance together with its variation significantly reduces binary classification hopping as shown in the next section.

3. RESULTS

This section presents the parameters of the decision filter, to address the *hand-detection* problem in FPV videos. The presented results compare performances of the naive (frame by frame) SVM detector proposed in [6] and the filtered version presented in this paper. The Kalman filter is formulated as a kinematic model of the position enriched with the speed, and a sampling rate Δ_t. Equations (9) and (10) are the process and measurement model, respectively. The tuning of the filter and the model selection remain an open issue, but it is important to keep these steps independent of the testing videos to ensure fair comparisons. Regarding x_0 and P_0, we initialize them as $[1, 0]$ and $I_{2\times 2}$, respectively. However, it is well known that this initialization is not critical in the linear case and related values quickly stabilize [23]. Looking for a smooth state we define a small covariance in the process model.

$$\begin{bmatrix} f(F_k) \\ \dot{f}(F_k) \end{bmatrix} = \begin{bmatrix} 1 & \Delta_t \\ 0 & 1 \end{bmatrix} \begin{bmatrix} f(F_{k-1}) \\ \dot{f}(F_{k-1}) \end{bmatrix} + w_k,$$
$$w_k \sim \mathcal{N}(0, Q), \tag{9}$$
$$Q = 0.001 * \begin{bmatrix} \frac{\Delta_t^3}{3} & \frac{\Delta_t^2}{2} \\ \frac{\Delta_t^2}{2} & \Delta_t \end{bmatrix}$$

$$z_k = [1, 0] \begin{bmatrix} f(F_k) \\ \dot{f}(F_k) \end{bmatrix} + v_k,$$
$$v_k \sim \mathcal{N}(0, 1) \tag{10}$$

Our approach is evaluated on the UNIGE dataset. This dataset contains a set of FPV videos, carefully recorded to guarantee a good balance between frames with hands and without hands, and to offer challenging characteristics such as sudden illumination changes, camera motion and hand occlusions. The dataset was recorded using a *GoPro hero3+* head mounted camera with a resolution of 1280×720 *pixels* and 50 *fps*. In total, 20 videos of approximately 3.34 minutes each (10020 frames) were recorded, 4 per location, of which two contain only frames with hands and the remaining two contain only frames without hands. Table 1 summarizes the length of the dataset in seconds, and Table 2 shows some examples of positive and negative samples for each location.

Table 1: Length in seconds of each of the parts of the dataset

	Office	Street	Bench	Kitchen	Coffe Bar
Hands	493	434	430	442	434
No Hands	470	434	450	442	462
Total	1.236	1.107	1.155	1.177	1.217

To train the SVM, the frames of the dataset are divided in training and testing. The training frames are the result of sampling one frame per second from the dataset (2203 frames with hands and 2233 frames without hands) while the remaining part of the videos are used for testing. The videos, as well as the training frames are distributed for public use in the ISIP40 website[1]. To evaluate the performances of the DBN, one SVM per feature is trained using the training frames. Subsequently, each SVM is used to classify the whole dataset in a naive fashion first (Frame by Frame) and then using the filtering framework proposed in section 2. Table 3 compares the naive SVM detector and the filtered decision approach. On average the proposed method improves the number of true-positives by 8.6 percentage points, moving from 82.4% to 91%, and the true-negatives by 6.0 percentage units, changing from 82.5% to 88.4%. Note that none of the performances is reduced by the DBN. The best performing feature for *hand-detection* according to our experiment is HOG, validating the finding of [6].

Figure 2 shows the overall performance of the HOG-SVM classification for each of the locations of the dataset. Red curve shows the measurement z_k while the blue one represents the state estimate $\hat{x}_k[0]$. The horizontal axis is the decision threshold (which is zero

[1] [Dataset:] http://www.isip40.it/resources/UNIGEhands.zip

Table 2: Examples of the dataset frames.

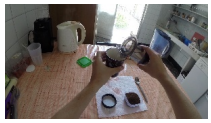
	Office	Street	Bench	Kitchen	Coffe Bar
Hands					
No Hands					

Table 3: Frame by frame vs. Kalman filter.

	$p_{gh}(1\|1)$		$p_{gh}(0\|0)$	
FEATURES	SVM	Filtered	SVM	Filtered
HOG	0.929	0.982	0.911	0.960
GIST	0.829	0.899	0.822	0.858
RGB	0.819	0.902	0.821	0.870
HSV	0.765	0.865	0.781	0.858
LAB	0.796	0.895	0.793	0.858
R+H+L	0.810	0.926	0.819	0.900
Average	0.825	0.911	0.824	0.884

in the general case). On the vertical axis at 4, 5, 6 (and -4, -5, -6) there are a) the ground truth, b) the decision of the naive HOG-SVM and c) of the filtered HOG-SVM, respectively. These decisions take positive values if hands are present and negative if not. The noisy movements of z_k confirm the dependence of the measurement to small changes in frames. As it is intended, the Kalman filter reduces the noise while preserving the trend of the detection. It can be noticed from the decisions of the HOG-SVM method that it is difficult to differentiate continuous segments of the video, with or without hands. This effect is the consequence of the measurement noise changing frequently the sign of z_k. Once the noise is reduced using the Kalman filter, the decisions stabilize and continuous segments appear. Particularly remarkable is the performance of the DBN modelling in the office and the bench sequences.

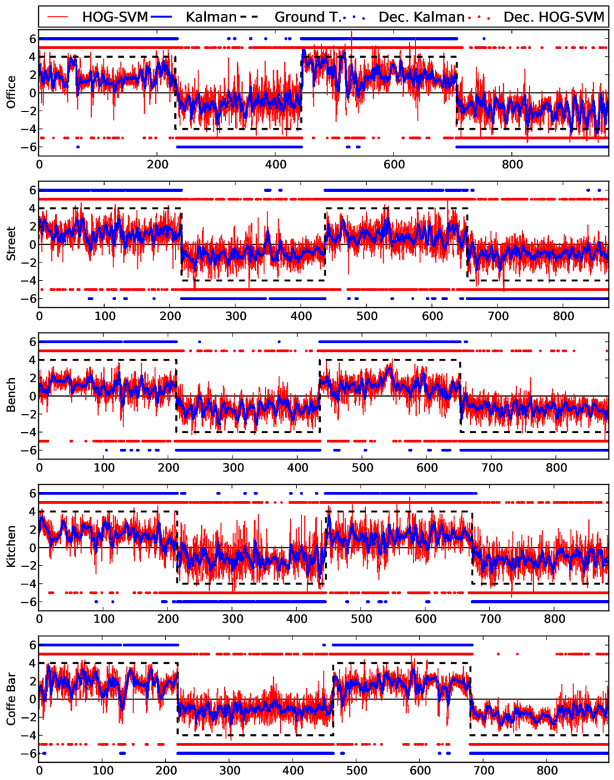


Fig. 2: Performance of the smoothing procedure in each location of the dataset.

4. CONCLUSION AND FUTURE WORK

This paper presented a filtering framework for frame-by-frame object detection smoothing in video sequences. Noise in the detection, mainly caused by small changes between frames, is reduced by filtering the decision process directly. The proposed method filters the signed classification distance $f(F)$ of the extracted features F from the decision boundary, estimated by a SVM, allowing to filter a single real valued variable (augmented with its variation) instead of a multidimensional vectors like HOG, GIST and color histograms.

The proposed method is evaluated in the hand-detection problem in First Person Vision, improving the classification rates of the state of the art by 8.6% and 6% for true-positives and true-negatives respectively. Presented results validate previous findings, where the combination HOG-SVM was stated as a good approach for hand detection.

As already noted, we think that filtering the signed distance $f(F)$ can give a feedback to the classifier on how well it is performing and allows to make the detection problem adaptive through such early self-awareness. This analysis is proposed as future research line, where the system will be able to modify its model according to the current performance.

Future research lines include an analysis of the impact of computation time required by different features on the filtering process: in a real time framework, the more the computation time, the less the actual processing frame rate; thus the less the temporal smoothness of consecutive features; consequently the more their need of being smoothed. Also, we have not discussed here the parameter tuning phase for the KF, nor more complex separation hyperplanes to be obtained from the SVM with the introduction of kernels. Eventually, the framework could be extend to a multi-class problem.

5. REFERENCES

[1] Paul Viola, Michael J. Jones, Daniel Snow, and Daniel Snow, "Detecting pedestrians using patterns of motion and appearance," in *In ICCV*, 2003, pp. 734–741.

[2] N Dalal and B Triggs, "Histograms of Oriented Gradients for Human Detection," in *Conference on Computer Vision and Pattern Recognition*. 2005, vol. 1, pp. 886–893, Ieee.

[3] P Dollár and C Wojek, "Pedestrian Detection: a Benchmark," in *Computer Vision and Pattern Recognition*, Miami, FL, 2009, pp. 304–311, IEEE.

[4] S Walk, N Majer, K Schindler, and B Schiele, "New Features and Insights for Pedestrian Detection," in *Conference on Computer Vision and Pattern Recognition*. June 2010, pp. 1030–1037, Ieee.

[5] Wen Gao, Xiaogang Chen, Qixiang Ye, and Jianbin Jiao, "Pedestrian detection via part-based topology model," *19th IEEE International Conference on Image Processing (ICIP)*, pp. 445–448, Sept. 2012.

[6] A Betancourt, Lopez M, M Rauterberg, and C.S. Regazzoni, "A Sequential Classifier for Hand Detection in the Framework of Egocentric Vision," in *2014 IEEE Conference on Computer Vision and Pattern Recognition Workshops*, Columbus, Ohio, June 2014, vol. 1, pp. 600–605, IEEE.

[7] ZW Kim and J Malik, "Fast Vehicle Detection with Probabilistic Feature Grouping and Its Application to Vehicle Tracking," in *Computer Vision*, Nice, France, 2003, pp. 524 – 531, IEEE.

[8] Srinivas Mukkamala, Guadalupe Janoski, and Andrew Sung, "Intrusion detection using neural networks and support vector machines," in *Neural Networks, 2002. IJCNN'02. Proceedings of the 2002 International Joint Conference on*. IEEE, 2002, vol. 2, pp. 1702–1707.

[9] P. and M. J. Jones, "Rapid object detection using a boosted cascade of simple features," in *IEEE International Conference on Computer Vision and Pattern Recognition, CVPR 2001*, 2001.

[10] Jerome Friedman, Trevor Hastie, and Robert Tibshirani, "Additive logistic regression: a statistical view of boosting," *Annals of Statistics*, vol. 28, pp. 2000, 1998.

[11] P. Kakumanu, S. Makrogiannis, and N. Bourbakis, "A Survey of Skin-color Modeling and Detection Methods," *Pattern Recognition*, vol. 40, no. 3, pp. 1106–1122, Mar. 2007.

[12] M Jones and J Rehg, "Statistical Color Models with Application to Skin Detection 2 Histogram Color Models," in *Computer Vision and Pattern Recognition*, Fort Collins, CO, 1999, vol. Jun, pp. 1–23, IEEE Computer Society.

[13] K Murphy, A Torralba, D Eaton, and W Freeman, "Object Detection and Localization Using Local and Global Features," *Toward Category-Level Object Recognition*, vol. 4170, pp. 382–400, 2006.

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

[15] Thomas P. Karnowski, I Arel, and D. Rose, "Deep spatiotemporal feature learning with application to image classification," in *Machine Learning and Applications (ICMLA), 2010 Ninth International Conference on*, Dec 2010, pp. 883–888.

[16] A Shashua, Y. Gdalyahu, and G. Hayun, "Pedestrian detection for driving assistance systems: single-frame classification and system level performance," in *Intelligent Vehicles Symposium, 2004 IEEE*, June 2004, pp. 1–6.

[17] H. Huang, J.J. Legarsky, S. Gudimetla, and C.H. Davis, "Post-classification smoothing of digital classification map of st. louis, missouri," in *Geoscience and Remote Sensing Symposium, 2004. IGARSS '04. Proceedings. 2004 IEEE International*, Sept 2004, vol. 5, pp. 3039–3041 vol.5.

[18] Zongju Peng, Gangyi Jiang, Mei Yu, Shihua Pi, and Fen Chen, "Temporal pixel classification and smoothing for higher depth video compression performance," *Consumer Electronics, IEEE Transactions on*, vol. 57, no. 4, pp. 1815–1822, November 2011.

[19] Lorenzo Baraldi, Francesco Paci, and G Serra, "Gesture Recognition in Ego-Centric Videos using Dense Trajectories and Hand Segmentation," in *Computer Vision and Pattern Recognition*, Columbus, Ohio, 2014, pp. 688–693, IEEE Computer Society.

[20] A Betancourt, P Morerio, C.S. Regazzoni, and M Rauterberg, "The Evolution of First Person Vision Methods: A Survey," *IEEE Transactions on Circuits and Systems for Video Technology*, pp. 1–1, 2015.

[21] A Betancourt, P Morerio, L Marcenaro, E Barakova, M Rauterberg, and C.S. Regazzoni, "Towards a Unified Framework for Hand-based Methods in First Person Vision," in *IEEE International Conference on Multimedia and Expo*, Turin, 2015, IEEE.

[22] T Hastie, R Tibshirani, and J Friedman, *The Elements of Statistical Learning*, Springer, 10 edition, 2009.

[23] Greg Welch and Gary Bishop, "An introduction to the Kalman filter," 1995.