Tracking both pose and status of a traffic light via an Interacting Multiple Model filter

Guillaume Trehard
IMARA Team
INRIA Rocquencourt, France

Evangeline Pollard
IMARA Team
INRIA Rocquencourt, France

Benazouz Bradai
Valeo Bobigny, France

Fawzi Nashashibi
IMARA Team
INRIA Rocquencourt, France

***Abstract*—Either for driver assistance systems or autonomous vehicles, detecting traffic lights (status and pose) is required when Intelligent Transport Systems go downtown. As detection algorithms could still have some misclassification on the traffic light status, this paper proposes a solution to nearly avoid this problem. An Interacting Multiple Model filter is used to track both the position and the status of a traffic light through the time and to increase traffic light recognition performances for automation purpose.**

I. INTRODUCTION

Advanced Driver Assistance Systems (ADAS) and autonomous driving gain nowadays an increasing interest in the automotive industry. Being able to help humans in their driving task could indeed lead to safer and more comfortable cars as well as more efficient road network. The poor invasive properties and the efficiencies of some of them (navigation system, park aided system...) have even encountered a true success in the public.

From early works in Intelligent Transport Systems (ITS), detecting traffic lights has been seen as a key point for assisted vehicles to go downtown [1]. Their status together with their positions are relevant information that could actually be detected in a passive or a non-passive way.

On the first hand, the non-passive methods are based on communication between the vehicle and the traffic light itself and are really efficient nowadays. Using Vehicular Ad-Hoc Networks (VANETS) [2] or even visible light communication [3], those systems enable fully managed intersection [4] but require special infrastructures and still need to be improved when it comes to dense traffic.

On the other hand, the passive systems are mainly based on vision techniques and use a front camera mounted in the vehicle as a sensor. Added to an *a priori* knowledge on the traffic light position in world reference (stored in a map), the camera output enables for example to detect its status by using a color segmentation algorithm [5], [6]. Region of interest in the video frame could also be deduced from road rules [7] in order to reduce the noise on a spot light detection. Another solution proposed in [8] is to train a Support Vector machine (SVM) to classify the color of a spot detected via a spot light detection algorithm. Vision characteristics such as Hough transform [9] could finally be used to extract the traffic light box from the image background.

The method used in this article is the one introduced by de Charette [10]. Coming out of the camera, spot lights are first extracted from the video frame with an algorithm based on the brightness property of traffic light spots. A template is then matched and evaluated for each spot by a series of basics operators such as a color validation for the switched-on spot or its halo. This template depends on the country and is a schematic way to describe a traffic light seen in an image. The output is a vector of traffic lights states composed of their position in the video frame, their radius and their status (Fig. 1).

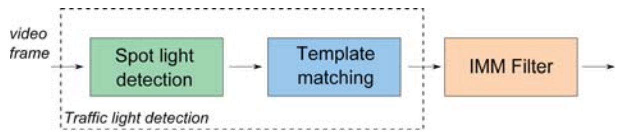

Fig. 1: The Traffic Light detection system and the IMM Filter

In addition of this detection process, few filtering methods have been implemented in order to enhance the output of a detection system alone. This filtering step is highly required to avoid the noise which affects both the pose and the status and which could lead to dangerous situation for a driver or an autonomous vehicle. A proper filter indeed enables to check the coherence of a traffic light track through the time and so to erase a lot of false detections. The classic way to do so is to use a *Constant Velocity* model in a Kalman filter to estimate the position of the traffic light in a frame. To filter the status output, a light sequence [11] or a threshold on track age [7] could be used as a simple model to avoid obvious misclassification.

The work of Nienhuser *et al.* is the closest to this article materials [8]. They have proposed to consider the status of a traffic light as following a Hidden Markov Chain to estimate the probability of a status to switch from one to another.

If all those techniques lead to interesting results, none of them consider the position and status evolution as the same process that need to be filtered. If the status could inform on the switched-on spot position, measuring this position can also reinforce the knowledge on the status so that both measurements are linked.

The proposed approach in this paper is so to consider the position and status estimation as a classic filtering problem.

Keeping the detection algorithm with its limits, the goal is then to improve the quality of its output by filtering its coherence through the time. Assuming a perfect association between a new measure and its track, the single-target tracking problem will be considered.

The context of this new filtering step is introduced in the first section of this article, a rigorous set up of an Interacting Multiple Model filter is made in section III and the performances of this proposition are shown in the last part.

II. Traffic Light Tracking

Since the traffic light recognition (TLR) field has its own specificities, this section proposes to highlight this peculiar context and set a practical background for the IMM algorithm introduction (Sec. III). The difference between the *continuous* evolution of the traffic light position and the *discrete* switches of its status is a fundamental part that requires particular attention.

A. Measure description

Focusing on the output of detection system proposed by R. de Charette [10]. The basic information available at iteration k is the position of the switched-on spot in the video frame $U_m^{spot}(k)$, its radius $r_m^{spot}(k)$ and the status of the traffic light $S_m(k)$. As seen in Fig. 1, this information will serve as measurement for the proposed filter.

$$Z_{TLR}(k) = \begin{pmatrix} Z \\ S_m \end{pmatrix}_k = \begin{pmatrix} U_m^{spot} \\ r_m^{spot} \\ S_m \end{pmatrix}_k \qquad (1)$$

The notation $Z(k)$ in eq. 1 highlights the fact that $U_m^{spot}(k)$ and $r_m^{spot}(k)$ actually have a *continuous* evolution for a given status $S_m(k)$. The status $S_m(k)$ is considered as a discrete random variable following a Hidden Markov Chain [8] and its realisations are part of a finite set depending on the country (This article will largely use the French example with $S \in \{red, green, amber\}$).

The limitations of the detection algorithm along with lighting and saturation problems in video frames [11] are responsible of a noise on the status S_m. This noise is independent on the position and radius part of the measure Z_{TLR} and is assumed to follow an uniform distribution. The error caused by a noisy realisation of S_m is named *false status error* and it then occurs with an unknown rate during the time a traffic light is visible by the camera (Fig. 2).

In addition, the vehicle which supports the camera is animated by pitch and roll that affect the video frame. Considering a camera with low distortions, the resulting noise is isotropic so that the pose $U_m^{spot}(k)$ and size $r_m^{spot}(k)$ are affected with a noise that will be modelled as a Gaussian distribution.

B. State to filter

It is worth to notice in Fig. 2 that if the switched-on spot of a traffic light has not a continuous evolution when the status changes, the position of the blackboard behind is supposed to evolve in a continuous way all along the time it is visible by the camera.

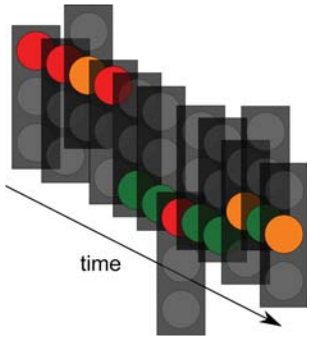

Fig. 2: Status errors on a traffic light evolution

This remark motivates the choice of tracking the blackboard position $U^{bb}(k)$ instead of the spot position $U^{spot}(k)$. The chosen state is then:

$$X_{TLR}(k) = \begin{pmatrix} X \\ S \end{pmatrix}_k = \begin{pmatrix} U^{bb} \\ \dot{U}^{bb} \\ r^{spot} \\ \dot{r}^{spot} \\ S \end{pmatrix}_k \qquad (2)$$

In order to link the positions $U^{bb}(k)$ and $U^{spot}(k)$, the idea of a traffic light template from [11] has been used. This template is a linear approximation of the position of the switched-on spot in the blackboard for each status (eq. 3). It could differ from a country to another and its inaccuracy could be covered by the noise on the evolution model (Sec. III-B).

$$U^{spot}(k) = U^{bb}(k) + r^{spot}(k).C_s \qquad (3)$$

where C_S is a constant vector depending on the status and describing the position of the switched-on spot in the blackboard, according to the chosen template (Fig. 3).

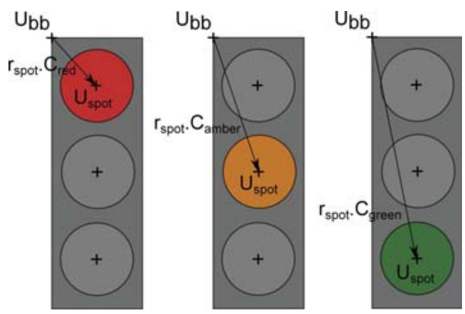

Fig. 3: Example of templates used for French traffic light

C. Linear system with Markovian switching coefficient

The state vector X_{TLR} and the measure vector Z_{TLR} thus contain respectively a part X and Z that follows a continuous evolution and respectively a discrete part S and S_m that takes realisations in a finite set.

In order to model the traffic light evolution in the picture and its growing speed, a classical *Constant Speed* model is applied on U^{bb} and r^{spot} and the status S can be considered as a discrete random variable.

If this *Constant Speed* model is well adapted for the situation in which the vehicle is coming front the traffic light and in a straight trajectory, it is worth to highlight a lack of representation when it comes to complex situations such as an approach in a curbed road or at an angle (*i.e.* in multi-lane scenarios). However, those cases are assumed to be covered by the noise on this model in the following study.

This complete evolution model has been described by Blom [12] as a linear system with a *Markovian switching coefficient* and the theoretical solution he presented was the Interactive Multiple Model algorithm (IMM). Numerous works have then proven the efficiency of this algorithm and it is now widely used in the literature, mostly in target tracking problems [13].

III. Interacting Multiple Model

The following linear system is first considered to represent the traffic light evolution and introduce the IMM algorithm.

$$\begin{aligned} X_s(k) &= F_s.X_s(k-1) + \nu(k) \\ Z_s(k) &= H_s.X_s(k) + \omega(k) \end{aligned} \tag{4}$$

with $s \in [1, n]$ and n the number of possible realisation of the status S. $\nu(k)$ and $\omega(k)$ are random variables respectively representing the white noise on the evolution and measure model. They are following Gaussian distribution : $\nu \sim N(0, Q)$ and $\omega \sim N(0, R)$ with Q and R the covariance matrices of those noises. The couple of matrices $(F, H)_s$ is actually depending on the s^{th} realisation of S.

A. Theoretical background

Presented in [12], the IMM algorithm is a sub-optimal approach to solve the problem of hypotheses merged in a linear system with *Markovian switching coefficients*. It is actually a first order approximation of the Full Hypothesis Tree estimator (FHT) which enables real time implementation by considering only current possible hypotheses [14].

The goal of IMM is to estimate the two first moments $\hat{X}(k \mid k)$ and $\mathbf{P}(k \mid k)$ of the posterior density $p[X(k) \mid Z(0:k)]$ by merging the posterior density $p[X(k) \mid S(k), Z(0:k)]$ of each model S where $X(k)$ is the realisation of X at time k and $Z(0:k)$ describes all the measurements Z from time 0 to time k.

Knowing the posterior probability of each model S at the precedent iteration $k-1$, $p[X(k-1) \mid S(k-1), Z(0:k-1)]$, the algorithm is divided in three steps:

- **Interaction:** Compute $p[X(k-1) \mid S(k), Z(0:k-1)]$ knowing the precedent model probabilities $\mu_s(k-1)$ and the model switching matrix ν_{ij}
- **Prediction:** $p[X(k) \mid S(k), Z(0:k-1)]$ is computed for each model using a Kalman filter with the evolution model presented in eq. 4.
- **Update:** $p[X(k) \mid S(k), Z(0:k)]$ is finally estimated with the new measure $Z(k)$.

The IMM filter then merges the posterior probability of each model $p[X(k) \mid S(k), Z(0:k)]$ using a Gaussian mixture. The obtained density is finally approximated as a Gaussian distribution to compute the posteriori density $p[X(k) \mid Z(0:k)]$.

The model probabilities $\mu_s(k)$ plays an important role in this process by weighting the impact of each model on the final estimated posterior density.

B. Evolution and measure matrices

As seen in section II-B, a *Constant Speed* model is used and a template enables to approximate the relation between $Z_s(k)$ and $X_s(k)$ (eq. 3). Using all these assumption, the evolution matrices F_s and measure matrices H_s of our system can be written as follow:

$$\begin{aligned} F_s &= \begin{pmatrix} 1 & \Delta t & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & \Delta t \\ 0 & 0 & 0 & 1 \end{pmatrix} \\ H_s &= \begin{pmatrix} 1 & 0 & C_i & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \end{aligned} \tag{5}$$

with Δt the time difference between two measures and C_s the constant introduced in eq. 3.

It is worth to notice that this particular example does not require different evolution matrices F_s for the realisations of S because the position of the blackboard does not depend on its status. The measure matrices H_s are the one which set this difference between the status. This way of using the IMM, even if it is fully included in the theory of the algorithm, has something original when most of practical applications consider one same measure matrix and several evolution matrices.

C. Status measurement

The model probabilities $\mu_s(k)$ used in the IMM process to evaluate the confidence of each model could also be seen as the probability of each status S, knowing the precedent probabilities $\mu_s(k-1)$ and the new measurement $Z(k)$. However, Z is not the only measure available at iteration k: the measured status S_m also gives an information on the status to estimate (Sec. II-A).

Since this status is independent of the the state X, the proposition here is to add a new step at the end of the IMM process to update the model probabilities $\mu_s(k)$ with the new measurement $S_m(k)$.

$$\mu_s^{corr}(k) = \frac{1}{\Gamma} p(S \mid Sm)(k).\mu_s(k) \tag{6}$$

with Γ a normalisation constant.

As seen in Sec. II-A, the status has been assumed to be affected by a uniform noise. An *a priori* on the false status rate τ_{fs} is then used to approximate $p(s \mid Sm)(k)$. This parameter could be seen as the confidence accorded to the traffic light detection system to detect the correct status and it will be added as follow:

$$\mu_s^{corr}(k) = \begin{cases} \frac{1}{\Gamma}.(1-\tau_{fs}).\mu_s(k), & \text{if } S = S_m(k) \\ \frac{1}{\Gamma}\frac{\tau_{fs}}{(n-1)}.\mu_s(k), & \text{else} \end{cases} \tag{7}$$

with n the number of possible realisations of S.

Knowing the measured status $S_m(k)$, Eq. 7 thus enables to update the model probabilities $\mu_s(k)$ by weighting them linearly, depending on τ_{fs}.

This change will impact the confidence of each status at iteration k and the estimated state at next iteration using the following relation:

$$\mu_s(k-1) = \mu_s^{corr}(k-1) \tag{8}$$

The knowledge about the status of a traffic light will so enhance its corresponding model probabilities without breaking the IMM process coherence.

D. Model switching probability

The model switching probability matrix ν_{ij} represents the probabilities to switch from a model to another. If a full parametrization could be done as in [8], three remarks needs to be considered when this matrix is set up:

- A switch between two status does not occur a lot while a car pass by a traffic light and a switch represents a really quick action when seen in perspective of all the traffic light track. Then, an important confidence must appear on the diagonal of the matrix.
- The status switches of a traffic light are ruled by a predefined logic that does not change a lot from one traffic light to another when the car stays in the same country. For example, the French cycle is always: *green, amber, red, green, amber....*
- Some non-detections could infer in the first two remarks so that the switching model must not be too rigid to be sufficiently robust.

As an example, the ν_{ij} matrix used in the following exemple (section IV) is shown in equation 9.

$$\nu_{ij} = \begin{pmatrix} 0.97 & 0.2 & 0.1 \\ 0.1 & 0.97 & 0.2 \\ 0.2 & 0.1 & 0.97 \end{pmatrix} \tag{9}$$

E. Initialization

As every statistic filtering method, the IMM algorithm is sensible to the quality of the initialization. A classical initialization for the components of Z is done by using the two first measures but the model probabilities are initialized with the same *a priori* on the false status rate τ_{fs} than in Sec. III-C.

$$\mu_s(0) = \begin{cases} 1-\tau_{fs}, & \text{if } s = S_m(0) \\ \frac{\tau_{fs}}{n-1}, & \text{else} \end{cases} \tag{10}$$

IV. Results

In order to evaluate the performances of this new algorithm, simulations and tests have been performed based on French rules ($S \in \{red, green, amber\}$). A track generator following the linear model presented above (eq. 4) has been designed and some tests on real data enabled to validate the approach presented above. A confusion matrix is used as tool to evaluate the performances of the solution.

The principal parameter of the results presented in this section is the false status rate. Introduced in Sec. II-A, it represents the rate of false status through the time for the same traffic light track and it will show that the IMM highly compensate its effects.

A. Simulations scenario

The scenario proposed in the following simulations is the one of a car approaching a traffic light with a straight trajectory. The switched-on spot position and radius have then been generated via the theoretical equations and parameters introduced in Sec. III. As explained in Sec. II-C, other scenarios such as curbed road could have been handled with wider noise on the model evolution (*cf.* Eq. 4).

The first status of the traffic light has been randomly chosen in the French set ($red, green, amber$) and two switches has been performed following the French cycles ($red \rightarrow green \rightarrow amber \rightarrow red...$) in the time of experiment.

As assumed in the measure description (Sec. II-A), a Gaussian noise has been added to the position and radius of the spot and the status measurement has been affected by a noise following an uniform density with a false status rate parameter.

By repeating it several times, this simulation process enabled to cover a large amount of error cases and tricky situations so that the parameters used in the filter were the same to deal with simulation or real data (except for the false status rate which was a tuning parameter in simulations).

B. Model probabilities evolution

Since the choice of a status output would be based on its corresponding model probability, the first interesting result remains in this probability evolution through the time. Fig. 4 shows an example of this evolution of μ_s through the time and for each status.

Despite a high false status rate ($\tau_{fs} = 30\%$ in Fig. 4), the proposed result shows that the filter can keep a stable and correct output. Moreover, when the switch from one status to another occurs in the simulation, the corresponding model probabilities instantly switch too.

It has to be enhanced that those results are possible because the IMM algorithm does not take into account only the status information but the switched-on spot position too. It is indeed the combined information of the switched-on spot position in the image and the status that enables such results. The specific switch from a status to another is indeed tracked by the IMM because of the template introduced in Sec. II-B and

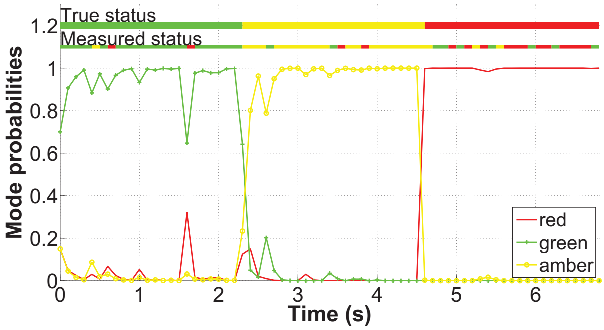

Fig. 4: Exemple of model probabilities evolution for $\tau_{fs} = 30\%$

when it occurs, it is validated by the status measurement. This combination leads to such robust results as the one in Fig. 4.

C. Position and size estimation

If the status is the most important information of a traffic light, its position in the image and size could lead to approximate the remaining distance between the vehicle and the traffic light and is so very useful too. Assuming both the intrinsic and extrinsic parameters of the camera known and by using an *a priori* on the size of the traffic light tracked, the pin-hole model could indeed enable to compute a distance to the traffic light. Even if such method is not presented here, it can be found in the literature [15].

Moreover, the IMM exactly aims at estimating properly this position and size. Despite the *Constant speed* model used (Sec. 4), it even appears that the filter presents a correct robustness to noise on the position U_m^{spot} and the size r^{spot}. Fig. 5 shows a result of such a filtering process.

An interesting result in Fig. 5 is that the quadratic error left on the estimated v coordinate of U^{spot} is nearly negligible. The template used in the measurement matrix H and presented in Fig. 3 indeed differs from one status to another only with this v coordinate. Thus, the measured status introduced in the IMM filter (III-C) adds an information on this coordinate so that its estimation remains better than a position tracking algorithm alone.

D. Monte Carlo validation

In order to evaluate performances of the algorithm to classify correctly the traffic light status, the track generator has been run 5000 times and each generated measures has been filtered through the IMM. The status to give as an output S_{output} of the IMM has been arbitrary set to the one with the maximum probability and the results are presented in a confusion matrix such as the one shown in Fig. 6. It is an efficient tool used in classification to evaluate the precision, the recall and the accuracy of a classifier.

Fig. 6 shows the performances of the IMM for the particular false status rate of 30% on the output of the detection algorithm. This false rate status means that the detection system alone would have a theoretic precision, recall and accuracy

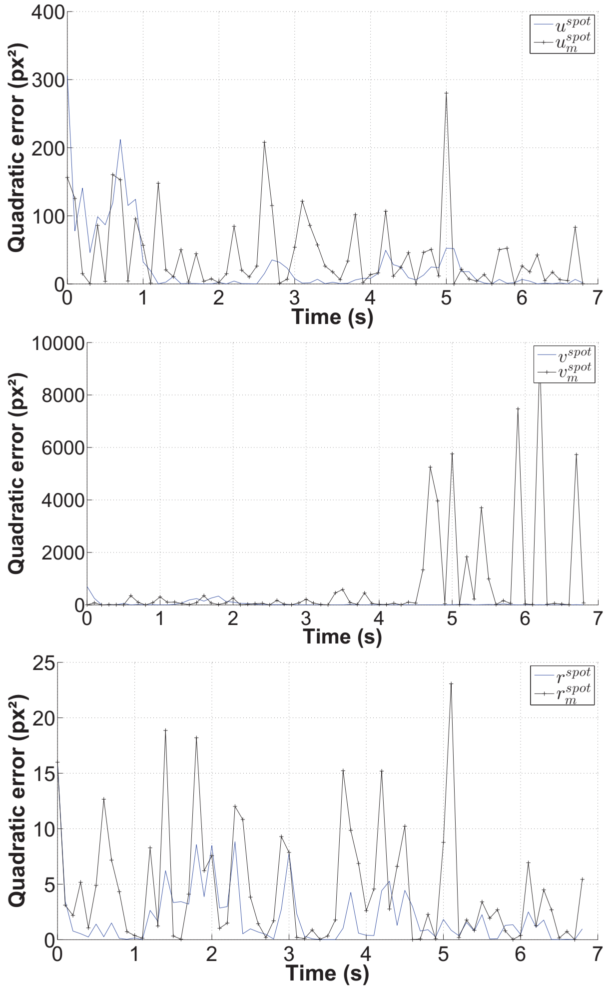

Fig. 5: Quadratic errors on U^{spot} and r^{spot} compare to their corresponding measure

		Estimated status			
		red	green	amber	recall
Ground truth	red	98.4%	1.5%	1.9%	96.6%
	green	0.8%	97.3%	1.0%	98.1%
	amber	0.8%	1.2%	97.1%	98.0%
	precision	98.4%	97.3%	97.1%	97.6%

Fig. 6: Exemple of a confusion matrix for $\tau_{fs} = 30\%$

of 70% so the improvements afford by the IMM are really impressive.

In the shown result, either the recall, the precision or the accuracy are indeed highly improved compared to the output of the simulated detection algorithm (from 26.6% and up to 28.4% for the red status precision).

E. Influence of the false status rate

Going further than the example shown in Sec. IV-D, the plots in Fig. 7 enhance the influence of the false status rate on accuracy, precision and recall.

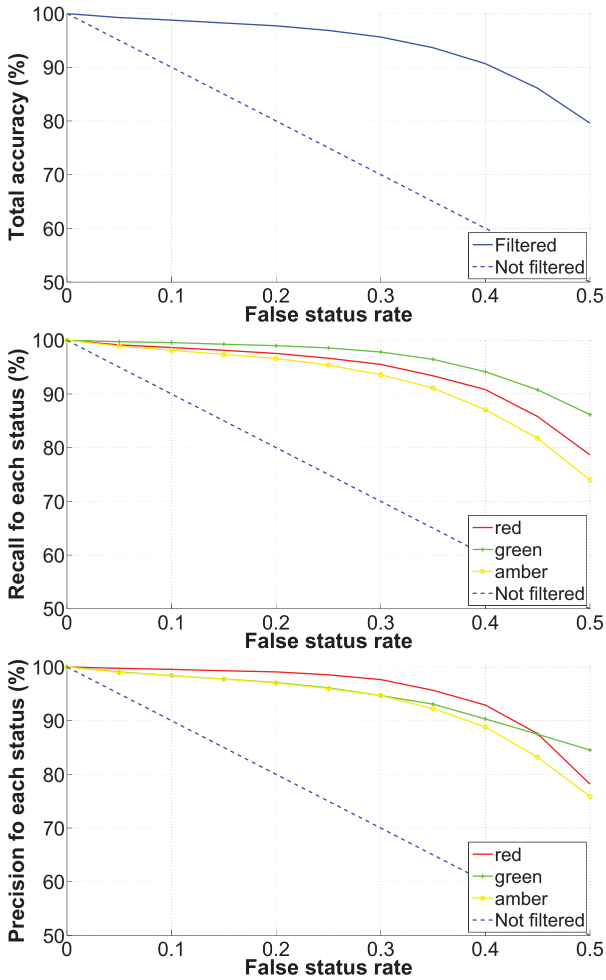

Fig. 7: Influence of the false status rate on accuracy, precision and recall

Even if an obvious result is that these parameters are decreasing along with the false status rate augmentation, It is worth noting they definitely do not follow the same slope as the output of the detection system alone (the blue dashed line in Fig. 7). As a consequence, the worth would be the detection output, the more useful would be such an algorithm.

These plots could also be seen as characteristics of the IMM algorithm and as a tool to design a complete traffic light recognition system. Evaluate how the filter can compensate the detection lacks of precision is indeed an important information.

F. Distance to the traffic light

If the results shown in Fig. 7 are encouraging, a small gap still need to be crossed to reach performances closer to 100%. However, these results are plotted for the whole experiences (from the time the traffic light is first viewed by the camera to the last time) and the influence of the distance on those track is actually worth to notice.

Since the noise on the switched on spot is independent of its size (Sec. II-A), the smaller (so the further) the traffic light is, the most it is affected by this noise. The coherence of the traffic light is indeed evaluated through the coherence of its blackboard evolution and the same noise on a small blackboard is more likely to make the algorithm estimate of a status switch.

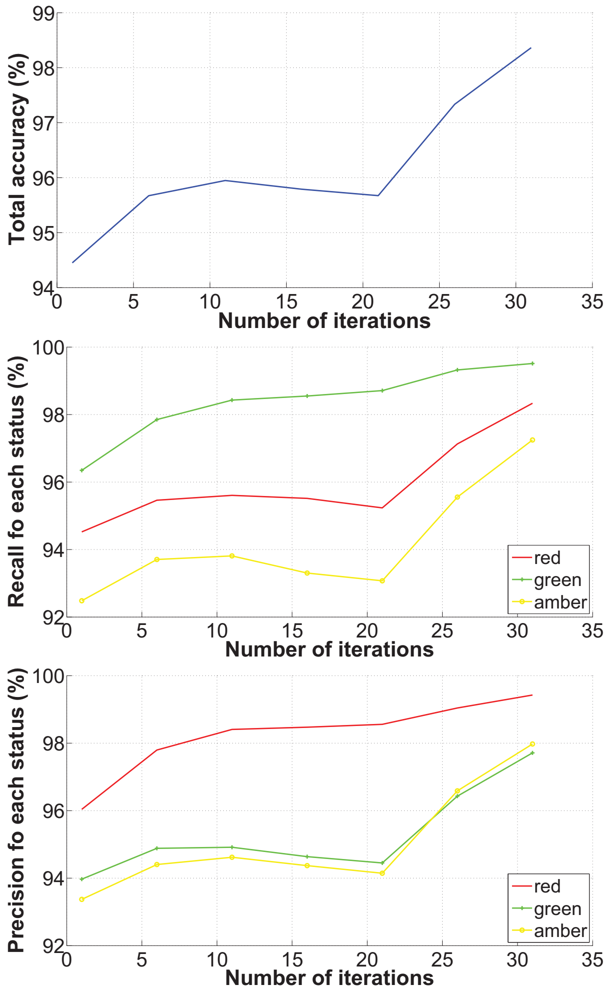

Fig. 8: Influence of the distance on the precision, recall and accuracy ($\tau_{fs} = 30\%$)

Fig. 8 is an illustration of this phenomena, the number of iterations in abscisse is closely linked to the distance left to the traffic light in the simulation scenario and all the plots show increasing performances when the traffic light is seen bigger (and so closer).

Thus, the confidence on the filter output is increasing when the car becomes closer to the traffic light and a strategy could be set up to give an output only for a minimum size and then increase the algorithm performances.

G. Validation on real data

In order to validate the theoretical results shown in this section, a validation step have been performed on real data. An example is presented in Fig. 9. The last frame of a video recorded on a French road is shown with the full measure track plotted on it. For the need of this article, the association between a measure and this track have been validated by an operator and all the false alarms have been removed to validate the single target tracking approach.

Fig. 9: Result of the algorithm on real data
top: Track history plotted on the last frame
bottom: Model probabilities for each status through the time

The same parameters as those used in the simulation have been used and the traffic light status is correctly estimated over the entire sequence. It is worth to notice that the first measure is a status error ($amber$ instead of $green$) and that this is corrected right after the second iteration. Moreover, the curvature of the road and the vehicle trajectory leaded to an evolution of the traffic light in the image which do not perfectly follow the *Constant Speed* model assumed in Sec. II-C. However, the output of the algorithm resists well to those inaccuracy.

V. Conclusion

The proposed algorithm has proven to be a robust solution which significantly improves the output performances of a classic traffic light detection system. By merging the position evolution and the status information of a traffic light through the time, it indeed appears that both states are increasing each other's accuracy and robustness.

This coherence validation has been shown to have better performances when the distance to the traffic light is shorter and a validation on real data has been performed. On this observation, this traffic light detection module can be used for autonomous driving.

The Bayesian formalism have enabled to properly evaluate the confidence of each status so that a planning algorithm could be set up to manage a more precise decision on the vehicle behaviour or on the information to give to a driver.

Because of the single target tracking hypotheses, the false alarms have been ignored in this article whereas it remains an important problematic in traffic light detection systems.

The multi-target problematic together with a system that could select the traffic light which concerns the driver or the autonomous car are still open and required field to be explored. Algorithm such as Multi Hypothesis Tracking (MHT) or Probability Hypothesis (PHD) could be worth to be tested.

References

[1] U. Franke, D. Gavrila, S. Gorzig, F. Lindner, F. Puetzold, and C. Wohler, "Autonomous driving goes downtown," pp. 40–48, 1998.

[2] V. Gradinescu, C. Gorgorin, R. Diaconescu, V. Cristea, and L. Iftode, "Adaptive Traffic Lights Using Car-to-Car Communication," pp. 21–25, 2007.

[3] N. Kumar, N. Lourenco, D. Terra, L. N. Alves, and R. L. Aguiar, "Visible light communications in intelligent transportation systems," pp. 748–753, 2012.

[4] K. Dresner and P. Stone, "A Multiagent Approach to Autonomous Intersection Management," *Journal of Artificial Intelligence Research*, vol. 31, pp. 591–656, 2008.

[5] N. Fairfield and C. Urmson, "Traffic light mapping and detection," in *IEEE International Conference on Robotics and Automation 2011 (ICRA)*, May 2011, pp. 5421–5426.

[6] J. Levinson, J. Askeland, J. Dolson, and S. Thrun, "Traffic light mapping, localization, and state detection for autonomous vehicles," pp. 5784–5791, 2011.

[7] M. Diaz-Cabrera, P. Cerri, and J. Sanchez-Medina, "Suspended traffic lights detection and distance estimation using color features," pp. 1315–1320, 2012.

[8] D. Nienhüser, M. Drescher, and J. M. Zöllner, "Visual State Estimation of Traffic Lights using Hidden Markov Models," in *IEEE Intelligent Transportation Systems, 2010 (ITSC)*, Madeira, Portugal, Sep. 2010, pp. 1705–1710.

[9] M. Omachi and S. Omachi, "Detection of traffic light using structural information," pp. 809–812, 2010.

[10] R. de Charette and F. Nashashibi, "Real time visual traffic lights recognition based on Spot Light Detection and adaptive traffic lights templates," pp. 358–363, 2009.

[11] R. de Charette, "Vision Algorithms for Rain and Traffic Lights in Driver Assistance Systems," Ph.D. dissertation, Mines ParisTech, 2012.

[12] H. A. P. Blom and Y. Bar-Shalom, "The interacting multiple model algorithm for systems with markovian switching coefficients," *IEEE Transactions on Automatic Control*, vol. 33, no. 8, pp. 780–783, Aug 1988.

[13] E. Mazor, A. Averbuch, Y. Bar-Shalom, and J. Dayan, "Interacting multiple model methods in target tracking: a survey," pp. 103–123, 1998.

[14] X. R. Li and Y. Zhang, "Multiple-model estimation with variable structure. Part V: Likely-model set algorithm," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 36, no. 2, pp. 448–466, 2000.

[15] J. Miura, T. Kanda, and Y. Shirai, "An active vision system for real-time traffic sign recognition," pp. 52–57, 2000.