**Preprints of the 19th World Congress**
**The International Federation of Automatic Control**
**Cape Town, South Africa. August 24-29, 2014**

# A Hidden Markov Model based transitional description of camera networks ⋆

**Riccardo Lucchese** * **Angelo Cenedese** * **Ruggero Carli** *

* *Department of Information Engineering, University of Padova, Italy.*

**Abstract:** We consider the problem of building a transitional model of an initially uncalibrated camera network. More specifically, we discuss a Hidden Markov Model (HMM) based strategy in which the model's state-space is defined in terms of a partition of the physical network coverage. Transitions between any two such states are described by the distribution of the underlying Markov Process. Extending previous work in (Cenedese et al., 2010), we show how it is possible to infer the model structure and parameters from coordinate free observations and we introduce a novel performance index for model validation. We moreover show the predictive power of this HMM approach in simulated and real settings that comprise Pan-Tilt-Zoom (PTZ) cameras.

Keywords: Graph models for networks; Sensor networks; Camera networks; Hidden Markov Models; Model Identification; Model Optimization.

## 1. INTRODUCTION

Camera networks play an increasingly important role in the modern society. The ubiquity of vision-like sensors is apparent in many contexts were they are widely employed to operate a variety of professional and recreational tasks. In a distributed scenario, the knowledge of the topology of the network by the acting agents is of extreme importance: the agents have to share information locally and need to cooperate with neighbors to attain the global performance. However, in situations where hundreds of nodes are spread across the environment it is practically impossible to have a manual setup of the network, and at the same time it is not convenient to rely on a topology that is a priori defined during the design phase. There is therefore the necessity of learning the topology of the network after it has been deployed.

A relevant problem in this context is to understand how the sensor scene relates to the environment. In the case of camera networks, the sensor scene is meant as the whole of visibility areas obtained through the union of the cameras's fields of view. As a practical example, consider the control room of a surveillance network where hundreds of cameras monitor a vast indoor or outdoor area with Pan-Tilt-Zoom (PTZ) or fixed cameras. In this context, once an event has been detected in a specific area, it would be desirable to predict which are the cameras that can provide information on the same event in the near future. Loosely speaking the problem is that of building a transitional model over the environment to be monitored, namely, to split the environment into sensor areas and to assign probabilities describing the possibility of an event to move from one area to another.

We refer to this problem as the *graph building problem* (Cenedese et al., 2010), since the solution is provided in terms of a graph where each node represents a sensor area and the edges stand for the admissible physical transitions between them. In addition a weight is assigned to each edge accounting for the probability of the corresponding transition.

Here we specifically consider the problem of building a transitional model of an initially uncalibrated camera network deployed in a realistic experimental setting. The construction of the graph and the estimation of the transitional probabilities is achieved by exploiting camera observations taken during the calibration phase of the system. In this phase, information on the network is generated by letting a known target move across the scene. The identification problem is then cast within the mathematical framework of Hidden Markov Models (HMMs) and the contribution of the paper extends previous work in (Cenedese et al., 2010) along three directions:
- first, we provide a richer description of areas which are not covered by any camera (the so-called null states);
- then, we introduce a novel performance index related to the HMM predictive power;
- finally, we provide performance results for both simulated and experimental settings that comprise PTZ cameras. These results show a superior behavior of the novel performance index with the respect to the one considered in (Cenedese et al., 2010).

*State of art:* The estimation of graphical models is a popular research subject (see, e.g., (Jordan, 1999; Bishop, 2006)) where the problem of fitting and selecting models on the basis of data is tackled for example by resorting to graph based structures (Markov random field) and to the formulation of a maximum likelihood optimization problem with a regularization term.

In the context of camera networks, a variety of tracking models with graph structure have been studied, see, e.g., (Javed, 2003; Zou et al., 2007; Farrell and Davis, 2008) and also the general discussion in (Mavrinac and Chen, 2013).

⋆ The research leading to these results has received funding from the European Community's Seventh Framework Programme under agreement FP7-ICT-257462 HYCON2 Network of excellence.

However, the focus is usually put either on statistical methods to discover relations between non-overlapping views or on the definition of classifiers to be used for reconstructing target trajectories.

Differently, (van de Camp et al., 2009) considers a HMM based approach: under the assumption that the coverage topology is known (or can be estimated as a preliminary step) a HMM is learned for each person moving throughout the scene and is then used to predict movement. A similar approach is discussed in (Vasquez et al., 2009) where the authors incrementally learn a Growing HMM. In (Marinakis and Gregory, 2009) a Markov process is used to infer the trajectories of multiple targets simultaneously present in the scene; the so learned process's parameters provide a description of the camera network in terms of a set of disjoint states and the corresponding transition probabilities.

*Organization of the manuscript:* Sec. 2 lists the assumptions made and introduces the notation used throughout this paper. Sec. 3 describes the HMM based identification procedure. Sec. 4 shows performance results in both simulated and experimental settings. Sec. 5 collects final remarks and future directions.

## 2. ASSUMPTIONS AND NOTATION

Throughout this paper we consider the following environmental model. The camera network operates in a bounded 2D domain $\mathcal{D} \subset \mathbb{R}^2$ and occluding objects in the scene are modeled as static 2D shapes with piece-wise linear boundary: the object's "walls".

The network comprehends $m$ cameras labeled by the indices $1, \dots, m$. A "fixed" camera has position and orientation both constant in time. PTZ units, instead, can control their angular position within certain pre-determined angular limits. The (un-occluded) Field-Of-View (FOV) of camera $i$, $\mathcal{F}_i \subset \mathcal{D}$, is modeled as a circular sector with origin at the point $\mathbf{0}_i$. The *coverage* of camera $i$, $\mathcal{C}_i \subseteq \mathcal{F}_i$, is defined as the subset of points of the corresponding camera FOV which are not occluded in the view of that camera. Since all walls are occluding it follows that a point $x \in \mathcal{F}_i$ is also in $\mathcal{C}_i$ if and only if the segment $\overline{\mathbf{0}_i x}$ does not intersect any wall in the environment. For a PTZ camera, both FOV and coverage sets are time-dependent. We define the domain of the camera with index $j$, $\mathcal{D}_j$, to be the conical sector obtained as the union of all the FOVs that are plausible for $j$.

Targets are point-like entities that are free to move in $\mathcal{D}$ on any trajectory, $\psi : \mathbb{R} \to \mathcal{D}$, that is a continuous function of time and such that its image does not intersect any wall in the environment (physical feasibility). A target with trajectory $\psi$ is said to be *detectable* by camera $i$ at time $t$ if $\psi(t) \in \mathcal{C}_i$.

It is assumed that no a priori geometrical information on the environment is available to the network and that cameras cannot rely on self-localization techniques. During the calibration phase, topological information is generated by letting a known target move across the scene. The corresponding observational datum at time $t$ is a binary ordered tuple, $\mathcal{O}_t \in \{0,1\}^m$, obtained by aggregating detection events from all sensors in the network. The $i$-th element of $\mathcal{O}_t$ is set to 1 if the target is detectable at $t$ by camera $i$, and is zero otherwise. We denote by $\mathcal{O}_1^T = (\mathcal{O}_{t_1}, \dots, \mathcal{O}_{t_T})$ a (time-)ordered T-tuple of observations. We will abuse this notation by writing $\tau$ in place of $t_\tau$ and $\mathcal{O}_\tau$ in place of $\mathcal{O}_{t_\tau}$.

In Sec. 4 we focus our analysis on the prototypical outdoor environment depicted in Fig. 1(a). The scenario mimics an actual deployment where a rectangular area, comprising two buildings, is surveilled by fixed (cameras 5 and 6) and PTZ cameras (cameras 1-4): such a model comprises both overlapping and isolated camera domains and occlusions, relating the network's sensors in a non-trivial way. A few remarks on the setting are in order. In the figure, gray lines starting from the cameras' origins denote the FOVs for fixed cameras and patrolling limits for the PTZ units. The FOV of each PTZ camera has been set to 10° degrees while the maximum panning speed has been limited to ten degrees per second in absolute value.

*Remark:* We stress that here we are not interested in solving the computational vision problem of *how* the camera sees the object of interest, but we are rather interested in *if* the camera sees it. We are thus considering a simplified framework and also do not deal explicitly with communication, synchronization and other issues.

## 3. THE HIDDEN MARKOV MODEL APPROACH

We now proceed to discuss how it is possible to infer a HMM description of the network in the framework outlined in Sec. 2. Let us recall that a HMM is a Markov process in which the system state is not directly observable, it is *hidden*, while an output, dependent on the state, is directly accessible and can be used to infer the system's trajectory. The basic elements of a HMM are (Rabiner, 1989; Rabiner and Juang, 2003)

- the system state space, $\mathcal{S} = \{S_1, S_2, \dots, S_N\}$; in the following we let $q_t$ denote the state at time $t$.
- the output alphabet $\mathcal{V} = \{v_1, v_2, \dots, v_M\}$; the elements of $\mathcal{V}$ are called observation symbols.
- the state transition probability distribution, the matrix $A \in \mathbb{R}^{N \times N}$ whose elements are given by the probabilities
$$a_{ij} := \mathbb{P}\left[q_{t+1} = S_j \mid q_t = S_i\right] , \quad 1 \le i, j \le N .$$
- the observation symbol probability distribution, the matrix $B \in \mathbb{R}^{M \times N}$ with entries
$$b_j(v_i) := b_{ij} := \mathbb{P}\left[O_t = v_i \mid q_t = S_j\right] ,$$
when $1 \le i \le M$ , $1 \le j \le N$.
- the initial state distribution (i.e. at $t = 1$), $\pi \in \mathbb{R}^N$
$$\pi_i := \mathbb{P}\left[q_1 = S_i\right] , \quad 1 \le i \le N .$$

A HMM with state space $\mathcal{S}$ is denoted in compact form by the tuple $\lambda = (A, B, \pi)$.

In the formulation of (Cenedese et al., 2010), the process's parameters are estimated following a two-step approach:

1) Initial structure discovery: static observations are used to establish a strong correspondence between the network's coverage cells and the HMM state space. The distributions $A$ and $B$ are inferred from the same data.
2) Parameter re-estimation and discovery of additional states: the initial HMM formulation is adjusted by an

iterative re-estimation procedure. By exploiting also the dynamics of the to-be-learned (output) trajectory some states are selected for a splitting procedure that reveals initially "hidden" states.

In the remainder of the paper we assume that a string of observations, say $\mathcal{O}_1^T$ for some $T > 0$, is available. We note that, in this effort, we focus on the problem of estimating the transition probabilities disregarding all self-loops. We thus explicitly discard any information on permanence times so that $\mathcal{O}_t \neq \mathcal{O}_{t+1}$ for all $t = 1, \ldots, T-1$.

### *3.1 Initial Structure Discovery*

The initial structure of the HMM is inferred starting from a coverage overlap model of the network, i.e. an abstract (coordinate-free) topological description of the (physical) camera network (on this topic see, e.g., Mavrinac and Chen (2013) and references therein). More precisely, a coverage overlap model establishes a set of distinguishable states in terms of coverage overlaps (or equivalently, in terms of the overlaps-induced coverage *cells*) and a set of relations, so that any two such states are related if there exist a physically admissible transition between them. Note that the information contained in this graph-like description is mapped in a natural and lossless way to HMMs.

We now notice that each binary observation $\mathcal{O}_t$ represents a signature of the states. More specifically, different observations reveal that the corresponding states of the system are distinguishable. Following an Ockham's razor interpretation of the available data we are thus lead to the following scheme. For each unique observation $\mathcal{O} \in \mathcal{O}_1^T = \{\mathcal{O}_1, \ldots, \mathcal{O}_T\}$, we introduce a new state, so that the HMM's state space is effectively defined by a one-to-one mapping from the set of coverage cells that have been discovered during calibration. Similarly, the set of admissible transitions constraining the Markov process is inferred from the set of transitions in $\mathcal{O}_1^T$.

Since coverage areas may not be connected, additional care must be taken with respect to how blind areas are modeled. To this aim, we endow the HMM with fictitious *null states*: the target is in a null state at time $t$ if its not detectable by any sensor in the network at time $t$. As in Cenedese et al. (2010), this situation can be formally described by introducing a single null state, that accounts for the whole unmonitored area and corresponds to null observations of the kind $\mathcal{O}_t = 0$. This approach leads, in general, to a loose description in the sense that the inferred model allows paths that are not admissible in the physical environment. In this paper, we propose a novel approach that estimates multiple null states each corresponding to a different blind area between some non overlapping camera coverages. In the new scheme, the discovery of null states is performed by scanning all the triplets in the observational string and by introducing a new state when a transition through an unmonitored area is witnessed. More precisely, we associate to the string $\mathcal{O}_t^{t+2} = (a, 0, b)$ for some $a, b \in \{0,1\}^m$, a unique null-state corresponding to the blind area in-between the coverage cells associated to $a$ and $b$. We notice that here we just wish to enable a formal description of the proximity relations between coverage cells. In this approach, two observations $(a, 0, b)$, $(b, 0, a)$, lead thus to the definition of the same null state.

Figs. 1(b)-1(c) show the coverage models obtained through the above procedures for the network of Fig. 1(a). We labeled each node with the identifier of the corresponding coverage cell: a label '1-4' denotes the cell given by the intersection of FOVs (or domains if a PTZ unit is involved) from cameras 1 and 4, i.e. $\mathcal{F}_1 \cap \mathcal{F}_4$. The symbol '$\sim$' stands for a transition to or from an unmonitored area. We notice that the quality of this initial description is influenced by the number and spatial distribution of the detection events that have been considered. If, as in this case, the network comprises also PTZ cameras, there is the additional difficulty that small coverage overlaps become increasingly difficult to map.

Assume then that the initial HMM state space has been determined. A unique observational symbol is now associated to each coverage cell. All null-states, instead, emit the same symbol 0. The HMM's output dictionary is taken thus to be a subset of $\{0,1\}^m$ and we are left with the determination of the initial process parameters. To this aim, we note that an intuitive direction is to choose them so as to maximize the probability of observing $\mathcal{O}_1^T$, i.e.,

$$\lambda_0 \in \underset{\lambda=(A,B,\pi)}{\arg\max}\ \mathbb{P}\left[\mathcal{O}_1^T \mid \lambda\right] . \tag{1}$$

However, the optimization manifold is usually complex and no analytical way to infer such $\lambda_0$ is known (see, e.g., Rabiner (1989)). Here we choose to define a possibly sub-optimal initial model by exploiting the following frequentist estimates. Define for all $1 \leq i, j \leq N$

$$\mathcal{T}_i\left(\mathcal{O}_1^T\right) := \text{number of visits to state } S_i \text{ in } \mathcal{O}_1^T ,$$
$$\mathcal{T}_{i\to j}\left(\mathcal{O}_1^T\right) := \text{number of transitions } S_i \to S_j \text{ in } \mathcal{O}_1^T .$$

Let then for all $1 \leq i, j \leq N$ and $1 \leq h \leq M$

$$a_{ij} = \begin{cases} \dfrac{\mathcal{T}_{i\to j}\left(\mathcal{O}_1^T\right)}{\mathcal{T}_i\left(\mathcal{O}_1^T\right)} & \text{if } \mathcal{T}_i\left(\mathcal{O}_1^T\right) > 0 \\ 0 & \text{otherwise} , \end{cases}$$

$$b_{hi} = \begin{cases} 1 & \text{if } h = i,\ S_i \text{ is not a null state} \\ 0 & \text{otherwise} . \end{cases}$$

Moreover let the null states emit the symbol 0 with probability one. The first phase is concluded by modeling the unknown state prior with a uniform distribution

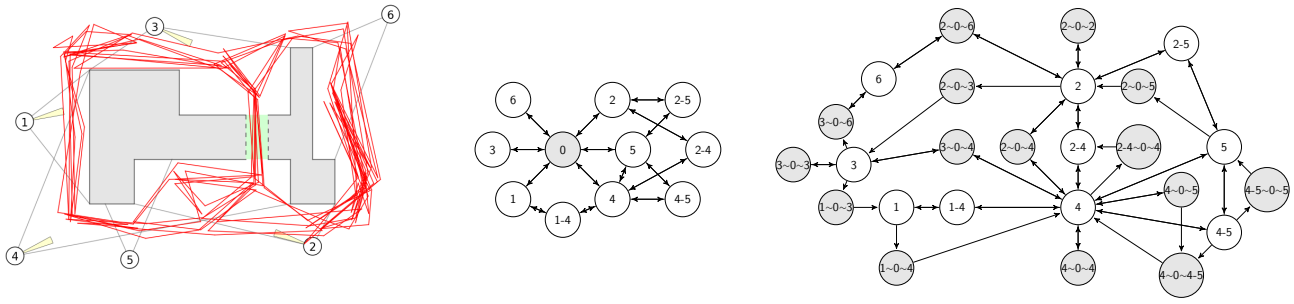
$$\pi_i = \frac{1}{N} , \quad 1 \leq i \leq N .$$

### *3.2 Parameter learning and node splitting identification*

The optimization manifold of problem (1) usually is characterized by multiple local maxima and, in fact, no technique can reach the optimum without recurring to exhaustive search. There exist, however, several gradient ascent methods that return a locally optimal choice of HMM parameters (see Rabiner (1989) and references therein), among which is the Baum-Welch algorithm (Baum et al., 1970; Welch, 2003). Starting from the current estimate $\lambda = (A, B, \pi)$ the latter provides a new model, say $\bar{\lambda} = (\bar{A}, \bar{B}, \bar{\pi})$, that satisfies (by construction) the property

$$\mathbb{P}\left[\mathcal{O}_1^T \mid \bar{\lambda}\right] \geq \mathbb{P}\left[\mathcal{O}_1^T \mid \lambda\right] , \tag{2}$$

with equivalence only if the optimization procedure has reached, through iterations, a local maximum. This optimization algorithm plays a central role in the identification scheme proposed here.




(a) (b) (c)

Fig. 1. (a): Schematic representation of the camera network considered in Sec. 4. The synthetic random trajectory used for performance evaluation in simulations is depicted as a red polygonal line. (b) and (c): The coverage overlap models, with a single null state and multiple null states respectively, inferred from a string of observations with $T = 250$.

Once a first instance of the HMM model has been retrieved, some of the states undergo a splitting procedure in order to disclose the presence of initially hidden states, which are replaced by two or more novel states. This procedure is devised exploiting the dynamics of the observation, taking into account also past and future states, as in Cenedese et al. (2010).

We observe that this splitting procedure only supports the identification of hidden states, which results in adding a new row and column to matrix $A$, while, concerning matrix $B$ only the number of columns is modified, as the number of observations does not change (specifically the column corresponding to the node $S_i$ splits into one for $S_i'$ and a new equal column for $S_i''$). Since the numerical values of transition probabilities to and from the new states are not known, a re-estimation pass (based on the observations) of the augmented matrices $A$ and $B$ is required. The new model is thus optimized trough the Baum-Welch algorithm.

### *3.3 Model validation*

To measure the performance of the identification procedure, it is introduced the following normalized correct prediction probability:

$$\eta = \sqrt[T]{\prod_{t=0}^{T-1} \mathbb{P}\left[\hat{y}_{t+1} = y_{t+1} \mid y_0^t\right]} = \sqrt[T]{\mathbb{P}\left[\mathcal{O}_1^T \mid \lambda\right]}, \qquad (3)$$

which corresponds to the (geometric) average probability of making the correct prediction for a specific sequence of observations $\mathcal{O}_1^T$ based on the identified model $\lambda$. This index can be interpreted as a sort of efficiency. In fact it is easy to see that $\eta \in [0, 1]$ for any observation sequence $\mathcal{O}_1^T$ and model $\lambda$, and clearly, higher values indicate better predictive capabilities. The strategy is thus to evaluate the performance index (3) for the validation datasets during the identification iterations and to use the feedback from the computed fits to stop the procedure as soon as overfitting is detected, i.e. when these validation fits start decreasing (see Cenedese et al. (2010)).

Unfortunately, an issue arises with the formulation in (3) since the index $\eta$ vanishes as soon as a validation trajectory $\mathcal{O}_1^T$ contains a transition that is not allowed by $\lambda$, independently of how well the model describes the rest of the sequence. Conversely, the performance indicator should remain informative even in presence of few low-probability transitions. In this perspective, we now introduce a different index $\rho$ related to how well the identified model can predict which sensor is able to provide information on the future position of the target.

Given a setting with $m$ cameras, define the following multifunction $\mathcal{S}(i) : \{0, 1, \ldots, m\} \to \mathcal{P}(\{0, 1\}^m)$ ($\mathcal{P}$ denotes the power set)

$$\mathcal{S}(i) = \begin{cases} \{\mathcal{O} \in \{0,1\}^m \ : \text{the } i\text{-th component of } \mathcal{O} \text{ is } 1\} \\ \qquad\qquad\qquad\qquad \text{if } i \in \{1, \ldots, m\}\,, \\ 0 \in \{0,1\}^m \qquad \text{if } i = 0\,. \end{cases}$$

For $i = 1, \ldots, m$, $\mathcal{S}(i)$ is the set of observational symbols that the network can generate when the target is detectable by camera $i$. $\mathcal{S}(0)$ instead represents a fictitious "sensor" with coverage extending over the whole unmonitored area. With this definition the Maximum Likelihood (ML) estimator of the future sensor is given by

$$\widehat{\psi}_{t+1} = \underset{i \in \{0,1,\ldots,m\}}{\arg\max}\ \mathbb{P}\left[\widehat{\mathcal{O}}_{t+1} \in \mathcal{S}(i) \mid \mathcal{O}_1^t\,,\ \lambda\right]\,. \qquad (4)$$

Then, the outcome $\widehat{\psi}_{t+1} = 0$, is interpreted as the prediction that the target will exit the monitored area at time $t + 1$, i.e. the model predicts that the most likely symbol at time $t + 1$ is 0. Let $\delta_t$ be the indicator function of the event $[\mathcal{O}_t \in \mathcal{S}(\widehat{\psi}_t)]$ and define the new performance index

$$\rho := \frac{\sum_{t=1}^{T-1} \delta_{t+1}}{T - 1}\,, \qquad (5)$$

namely the ratio of "correct" predictions over the number of transitions in the dataset. $\rho$ takes values in $[0, 1]$: $\rho = 1$ is achieved only if the model is able to perfectly predict all future sensors based on the past, while when $\rho = 0$ the model cannot correctly infer any of the predictions (4). In general, the closer is this index to one, the better are the HMM prediction capabilities.We adopt for $\rho$ the same stopping criterion described above.

*Remark:* The probability $\mathbb{P}\left[\mathcal{O}_1^t\,;\ \lambda\right]$ can become zero, e.g. if the validation trajectory contains an output transition that is not admissible in the HMM. In these cases the estimation procedure must be restarted, in the sense that when computing these estimates for times $\bar{t} \geq t$ past observations until $t - 1$ should be discarded, i.e.

$$\widehat{\psi}_{\bar{t}+1} = \underset{i\in\{0,1,\ldots,m\}}{\arg\max}\ \mathbb{P}\left[\widehat{\mathcal{O}}_{\bar{t}+1} \in \mathcal{S}(i) \;\middle|\; \mathcal{O}_t^{\bar{t}}\,,\ \lambda\right]\,, \quad \text{for } \bar{t} \geq t\,.$$

## 4. SIMULATIONS AND EXPERIMENTS

To analyze the performance of this HMM identification approach, we study the evolution of indices $\eta$ and $\rho$ (respectively (3) and (5)), for both synthetic and experimentally gathered observations. Moreover we compare the situations where the initial HMM structure is defined using either just one or multiple null states. To this aim, we focus to the specific case study of Fig. 1(a), where a network of both fixed and PTZ cameras performs the perimeter patrolling of a building structure or a generic area (see Sec. 2 for the details). This synthetic setup mimics the experimental testbed installed in the Autonomous Navigation Laboratory of the Dept. of Information Engineering of the University of Padova, in terms of disposition and parameters of fixed and PTZ cameras. For a first performance assessment, the target trajectories have been randomly artificially generated in the simulation scenario (as shown in Fig. 1(a)), while for a further experimental validation they are acquired from measurements in the real testbed.

The results of this analysis are summarized in Fig. 2, where the first two columns (subfigs. (a)-(b)-(d)-(e)) refer to the synthetic scenario whereas the third column (subfigs. (c)-(f)) is related to the experiments in the actual testbed. Also, in the first row (subfigs. (a)-(b)-(c)) it is considered a single null state to describe the situation when the target is not observable by the network, while in the second row (subfigs. (d)-(e)-(f)) the multiple null case is taken into account. In these plots, a solid black line indicates the performance of the training dataset while colored dashed lines refer to the performance relative to validation data. Loosely speaking, the performance indexes $\eta$ and $\rho$ measure the capability of the model to describe the observational data and the detected trajectories: the HMM model is obtained by learning the scene through the training data, and is then applied to different validation sequences to check its consistency and robustness.

We previously discussed how $\eta$ can vanish if the validation data cannot be explained in terms of the estimated HMM. A similar issue arises with the adoption of the node-splitting procedure: when a state is split, the following parameter re-estimation step might lead to a new model with a different set of admissible paths (of length 2). Presently this situation is not controlled by the procedure and may cause $\eta$ to evaluate at near zero values (for the validation datasets) as the identification algorithm is iterated. This effect is shown in Figs. 2(a) and 2(d).

We notice how, on the one hand, the training fits improve steadily as new hidden states are discovered. On the other hand, the evolutions of $\eta$ relative to the validation sequences are rather uninformative: they all eventually vanish giving little insight on the model's predictive power. In other words, in these cases $\eta$ fails to be a trustful indicator of performance. In Figs. 2(b) and 2(e), instead, we plot the evolution of performance index $\rho$, for the same synthetic observation strings. Similarly to the previous case, as the iterations occur, the evolution of the fits computed for the training string, sees $\rho$ first increasing and then remaining stable. Remarkably, Fig. 2(b), witnesses the same qualitative behavior even for the validation datasets. Moreover the strategy to exploit multiple null-states shows improved initial performance while the final models achieve roughly the same values for $\rho$. The identification procedure leads thus to HMMs that exhibit good overall predictive performance as measured through the $\rho$ index, which confirms to be more consistent than $\eta$ and can be thus employed as a performance indicator.

Finally, experiments have been conducted on the real camera network testbed. For practical reasons, here the underlying physical trajectory, from which the strings of observations are derived, is an eight shaped curve looping around the two buildings and the motion is more constrained. Example evolutions of performance index $\rho$ are presented in Figs. 2(c) and 2(f). Both figures show how the predictive performance of the HMM increases through iterations and, remarkably, the evolutions follow the same qualitative behavior of the simulated case. However, the final HMMs achieve better performance in this case. This is due to the fact that the constraints given by the physical testbed yield less randomness and variability if compared to the simulated scenario. I.e., the target movements employed to provide the observational strings in this case lead to intrinsically better predictable future positions.

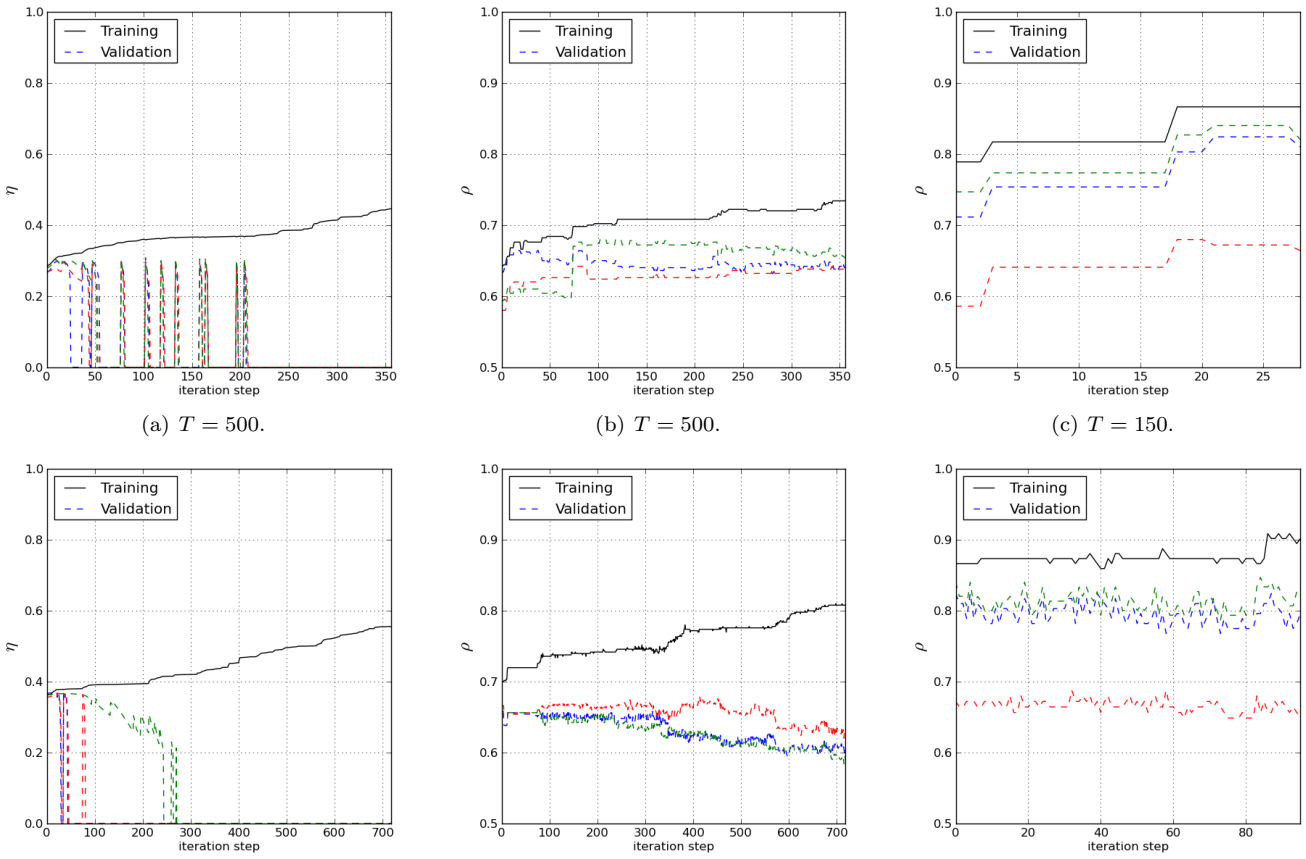
As for the single-null versus the multiple-null strategies, we observe that the latter shows better performances in all the considered case at the cost, though, of a higher complexity of the model, which poses a trade-off between prediction capability and model dimensionality.

## 5. CONCLUSIONS

In this work, we have considered the problem of building a transitional model of an initially uncalibrated camera network, extending previous work in (Cenedese et al., 2010). We have shown how it is possible to estimate a HMM based description of the network starting from coordinate free measurements of target activity and to validate our approach we have shown performance results from both simulated and real settings comprising fixed and PTZ cameras.

More specifically performance has been measured by two indices: the first related to the model's efficiency at predicting future outputs; the second instead is based on a predictor of which sensor is most likely to provide information on the future position of the target. Results of both synthetic and experimental nature exhibit a qualitatively similar behavior and show good overall performance; nonetheless, the second index, related to the prediction of the next useful sensor, appears more stable and shows a more consistent dynamics.

The approach appears thus as a promising tool for the estimation of transitional models in real camera networks, and hence this suggests to pay further attention to these techniques. More specifically we plan to better study the online estimation of the network structure, also considering the interplay between the learning of the model and suitable patrolling techniques that favor the former. Finally, a more extended validation in large testbed would be beneficial to get more insight into the performance of the procedure in real-world settings.




(a) $T = 500$. (b) $T = 500$. (c) $T = 150$.

(d) $T = 500$. (e) $T = 500$. (f) $T = 150$.

Fig. 2. Performance indices $\eta$ and $\rho$ as a function of the iteration step in the identification procedure. Evolutions in the upper row correspond to models where the unmonitored area is modeled by a single null node; in the lower row, instead, each blind area is mapped to a different null state. (a)-(b)-(d)-(e): synthetic observational data (target movement of Fig. 1(a)). (c)-(f): data gathered from the real camera network.

## REFERENCES

L. E. Baum, T. Petrie, G. Soules, and N. Weiss. A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains. *The Annals of Mathematical Statistics*, 41(1):164–171, 1970.

C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer, New York, 2006.

A. Cenedese, R. Ghirardello, R. Guiotto, F. Paggiaro, and L. Schenato. On the Graph Building Problem in Camera Networks. In *IFAC Workshop on Distributed Estimation and Control in Networked Systems (Necsys)*, pages 299–304, 2010.

R. Farrell and L. S. Davis. Decentralized discovery of camera network topology. In *2nd ACM-IEEE Int. Conf. on Distributed Smart Cameras*, pages 1–10, 2008.

O. Javed. Tracking across multiple cameras with disjoint views. In *9th IEEE Int. Conf. on Computer Vision (ICCV)*, volume 2, pages 952–957, 2003.

M. I. Jordan. *Learning in graphical models*. MIT Press, Cambridge, MA, 1999.

D. Marinakis and D. Gregory. Self-calibration of a vision-based sensor network. *Image and Vision Computing*, (27):116–130, 2009.

A. Mavrinac and X. Chen. Modeling Coverage in Camera Networks: A Survey. *International Journal of Computer Vision*, 101(1):205–226, 2013.

L. Rabiner and B. Juang. An introduction to hidden Markov models. *IEEE ASSP Magazine*, 3(1):4–16, 2003.

L. R. Rabiner. A tutorial on hidden Markov models and selected applications in speech recognition. *Proc. of the IEEE*, 77(2):257–286, 1989.

F. van de Camp, K. Bernardin, and R. Stiefelhagen. Person tracking in camera networks using graph-based bayesian inference. In *3rd ACM-IEEE Int. Conf. on Distributed Smart Cameras (ICDSC)*, pages 1–8, 2009.

D. Vasquez, T. Fraichard, and C. Laugier. Incremental Learning of Statistical Motion Patterns With Growing Hidden Markov Models. *IEEE Trans. on Intelligent Transportation Systems*, 10(3):403–416, 2009.

L. R. Welch. Hidden Markov Models and the Baum-Welch Algorithm. *IEEE Information Theory Society Newsletter*, 53(4), December 2003.

X. Zou, B. Bhanu, B. Song, and A. K. Roy-Chowdhury. Determining Topology in a Distributed Camera Network. In *IEEE Int. Conf. on Image Processing (ICIP)*, volume 5, pages 133–136, 2007.