Prediction in real-time image sequences is a key-feature for visual servoing applications. It is used to compensate for the time-delay introduced by the image feature extraction process in the visual feedback loop. In order to track targets in a three-dimensional space in real-time with a robot arm, the target’s movement and the robot end-effector’s next position are predicted from the previous movements. A modular prediction architecture is presented, which is based on the Kalman filtering principle. The Kalman filter is an optimal stochastic estimation technique which needs an accurate system model and which is particularly sensitive to noise. The performances of this filter diminish with nonlinear systems and with time-varying environments. Therefore, we propose an adaptive Kalman filter using the modular framework of mixture of experts regulated by a gating network. The proposed filter has an adaptive state model to represent the system around its current state as close as possible. Different realizations of theses state model adaptive Kalman filters are organized according to the divide-and-conquer principle: they all participate to the global estimation and a neural network mediates their different outputs in an unsupervised manner and tunes their parameters. The performances of the proposed approach are evaluated in terms of precision, capability to estimate and compensate abrupt changes in targets trajectories, as well as to adapt to time-variant parameters. The experiments prove that, without the use of models (e.g., the camera model, kinematic robot model, and system parameters) and without any prior knowledge about the targets movements, the predictions allow to compensate for the time-delay and to reduce the tracking error.

Keywords: Kalman filtering; Adaptive Kalman filter; Prediction; Target tracking; Mixture of experts; Neural networks; Neuro-control; Visual servoing.

1 Introduction

Tracking a moving object in a three-dimensional space counts among the basic applications of visual perception in robotics. However, in spite of a large amount of works, this apparently simple task remains a complex open problem, due to the system’s nonlinearities, to a large number of parameters, and to time-varying environments. Another well-known problem in real-time applications is the time-delay introduced in the arm control loop by the image grabbing and processing tasks. To cope with this constraints, a modular prediction scheme is proposed, composed of several Kalman filters and an artificial neural network (ANN).

Kalman filtering is a widely used estimation technique. The method however requires an exact knowledge of the model parameters for optimal behavior. Practically, only a simplified or approximated model is often available, valid in a limited workspace. These observations are true for the system’s dynamics and the noises properties inherent to the filter structure. Adaptive Kalman filters (AKFs) partially solve these problems. In addition to classical Kalman filtering, these adaptive filters are endowed with an
estimation process adjusting the unknown parameters of their structure. These techniques prove to be efficient for noise parameters estimation, but require modeling the system’s dynamics, assuming a time-invariant model. Kalman filtering without model derivation broadens its application field. We therefore introduce a filter with an adaptive state model that is able to compensate for system nonlinearities and changing properties, especially when the input space is very large. When the system’s behavior varies over time with abrupt changes or when the system’s parameters vary, we propose to use several noise models by using the modular learning scheme known as mixture of experts\(^1\).

The mixture of experts framework is a modular learning scheme composed of expert networks which all contribute to a global output and a gating network which mediates the expert networks outputs by learning their ability to give estimates close to a desired output for a given input value. In our approach, each expert is a state model AKF with different realizations of the unknown system parameters such as process and noise properties. The gating network of the mixture of experts performs online adaptation of its weights given individual filter estimates based on performances. This learning scheme weighs the estimates of the different AKFs and rates the state model adaptation for each AKF. The performances of the suggested adaptive filter structure are investigated by considering the modeling problems which limit the conventional Kalman filter (KF). As an application, the modular framework composed of AKFs is evaluated on the visual servoing of a three-degree-of-freedom robot manipulator. The robot arm has to follow a target consisting of a moving object in the 3-D space that is not associated to a model. A fixed stereovision sensor observes the scene, and an image processing system extracts target and end-effector coordinates. The modular AKF framework is then used for online prediction of the target’s and end-effector’s next positions to compensate for the delay introduced by the visual feedback loop, thus minimizing the tracking error. An artificial neural network update law provides online estimation of the Jacobian, and the control law is provided in the form of a linear local approximation technique. Robot control is independent of robot and camera configurations, and is able to reach both static and moving targets.

The expert networks and the gating network are all trained simultaneously, allowing for a real-time implementation. We show that with no \textit{a priori} knowledge about target movements – robot and camera models – the proposed modular network is able to make accurate predictions that are used by the neuro-controller to elaborate the robot’s joint angle commands to precisely follow a given moving target.

In this paper, we will first show that the classical Kalman algorithm can be replaced by a state model AKF which adapts its transition matrix and noises parameters overcoming thus model derivation problems. Follows a presentation of the mixture of experts principle and the analysis of AKFs organized in this framework. Finally, several examples illustrate the validity of our proposed predictive model independent visual servoing scheme.

2 Adaptive Kalman Filtering

2.1 Introduction

Consider the discrete state representation including the system’s evolution law and the output measurement relation, \(k\) denoting the discrete time index:
The KF is an optimal filter to estimate inaccessible variables $x_k$ from observed data $z_k$. A well-known limitation is the \textit{a priori} assumption about the statistics that describe the process noise, $w_k \sim (0, Q_k)$, and the measurement noise, $v_k \sim (0, R_k)$. In almost all applications, the system and measurement noises are assumed zero mean white noises to satisfy the requirement of the KF algorithm. If these assumptions cannot be held, or when full information about the system model is unavailable, a further modeling step will be the online estimation of the filter’s parameters. Thus, adaptive Kalman filtering consists of the following functional steps: estimating unknown parameters of the signal model, updating estimates within the KF structure, and the filtering itself.

The first work exploring this subject\textsuperscript{2} consisted in iteratively re-evaluating the process noise covariance $Q_k$ and the measurement noise covariance $R_k$. This approach mainly allowed to overcome the choice of appropriate noise properties. This is not straightforward. Four distinct methods are distinguished for covariance matrix estimation\textsuperscript{2}: Bayesian estimation, maximum likelihood estimation, correlation methods and covariance-matching techniques. Online parameter estimation has been extended to unknown deterministic inputs\textsuperscript{3}. These investigations were also confined to covariance matrix estimation. Indeed, they all consist in estimating $Q_k$ and $R_k$ by assuming that all other parameters are known, and in particular the system’s dynamics. They suppose an optimal filter, and if the innovation sequence is no more a white Gaussian sequence, $Q_k$ and $R_k$ are modified accordingly. For real-time applications, the implementation proves to be a difficult task: the methods must converge, they consist of several recursive loops, and consume some non negligible computational power.

Furthermore, the adaptation process can be designed to estimate transition matrix $A$, output matrix $C$, respectively in Eq. (2.1) and (2.2), and the optimal Kalman gain matrix $K_k$, using consecutive samples of observed data $z_k$.\textsuperscript{4} This approach is only valid under some strong assumptions which narrow its applications: the system must be single-input-single-output, minimum phase, completely controllable and observable, the system’s noise covariance must be identity, and state matrix $A$ must be nonsingular and stable.

A considerable amount of research has been carried out in the AKF area, but in practice it is often necessary to redesign the adaptive Kalman scheme according to the particular characteristics of each problem. The online estimation of noise properties does not overcome the modeling problem in a classical KF scheme.

### 2.2 State Model Adaptive Kalman Filter Predictions

We propose to use a KF with an adaptive state model that always fits the system around its current state, even if it is nonlinear. This also permits to cope with changes in the real system and with the absence of a parametric model. This new adaptive Kalman filter uses a transition matrix which depends on the two measurement vectors as will be demonstrated in the following. No \textit{a priori} knowledge is used to adapt the filter’s state space. To adapt the transition matrix $A_k$, the state equation (2.1) is simplified to a local linearized relation. This is done by making a couple of assumptions: firstly, the process must only be controlled by the stochastic part ($u_k = 0$ for all $k$) or the input must be treated as part of the disturbance, and secondly, the process noise must be centered, $E[w_k] = 0$. This leads to an estimated transition matrix that is only a function of the two
last state vectors: \( \hat{A}_k = x_k \hat{x}_k^{-1} \). Inverting a matrix by the particular solution approximated in the least mean squares sense, we can write: \( \hat{A}_k = x_k \left( \hat{x}_k^{-1} \hat{x}_k^{-1} \right)^{-1} \). The measurement equation (2.2) allows us to express the state vector with a given output \( z_k \). If we consider that the noise sequence \( v_k \) can be neglected over time \( \mathbb{E}[v_k] = 0 \), and if we know \( C_k \) at time \( k \), it is possible to have an estimation of its inverse \( C_k^{-1} \), thus for the two last iterations: \( x_k = \hat{C}_k^{-1} \hat{z}_k \), and \( x_{k-1} = \hat{C}_k^{-1} \hat{z}_{k-1} \). After simplification, the transition matrix can now be expressed with:

\[
\hat{A}_k = \left( \hat{C}_k \hat{z}_k \left( \hat{C}_k^{-1} \hat{z}_k \right)^T \right)^{-1} \hat{z}_k \hat{C}_k^{-1} \hat{z}_k^T.
\]

(2.3)

The previous expression depends on the dimensions of matrix \( C \in \mathbb{R}^{p \times n} \), with \( p \) the output vector's size and \( n \) the state vector's size. This matrix is in most cases non square. We will develop the transition matrix estimation for \( p < n \), but similar developments hold for \( p > n \) and \( p = n \).

Using the right pseudo-inverse to compute \( \hat{C}_k^+ \) and \( \hat{C}_{k-1}^+ \), and introducing \( G_k = \hat{C}_k \hat{C}_k^+ \in \mathbb{R}^{p \times p} \), expression (2.3) becomes:

\[
\hat{A}_k = \left( \hat{C}_k \hat{z}_k \left( \hat{C}_k^+ \hat{z}_k \right)^T \right)^{-1} \hat{z}_k \left( \hat{C}_k \hat{C}_k^+ \right)^T.
\]

(2.4)

The expression in brackets in (2.4) reduces to a scalar and can be easily inverted. It mounts up to a weighted dyadic vector product between the two last measures, \( z_k \) and \( z_{k-1} \).

Although \( \hat{A}_k \) is continuously adjusted through time, we cannot say that the process model is learned in the sense used in neural network applications. The value of \( \hat{A}_k \) is only valid in a restricted temporal area. This matrix depends on how the last two measures issued from the process are correlated.

The benefit of this scheme is to avoid the formal derivation of the system’s dynamics. This approach is stable, robust, and works best for a given type of statistics. To increase its adaptability capabilities (in the sense of its noise parameters), the filter is endowed with evolving covariance matrices. Thus, and to address more complex problems, with time-varying or various noise statistics, we propose to use a modular architecture composed of state model AKFs with different noise models.

3 The Mixture of Experts Framework

3.1 Basic Principle

In ANNs, the mixture of experts is a well-known modular approach for solving complex problems. This structure has been widely applied to different kind of modern problems such as classification, function approximation, prediction and control. The structure, shown in Fig. 1, consists of \( K \) supervised modules called expert networks, and an integrating module called a gating network that is a mediator among the expert networks. The modularity here is an implementation of the divide-and-conquer principle. This learning architecture was first proposed with simple linear regression processes in terms of expert networks, but different kind of neural networks or other estimation methods have since been integrated in this modular scheme.

At iteration \( k \), the output of the mixture is expressed by the weighted sum of the \( K \) expert networks, \( y_k = \sum_{i=1}^{K} g_{i,k} y_{i,k} \). The output \( y_{i,k} \) of the \( i \)th expert network is mediated with an activation function elaborated by the gating network:
The transformation of Eq. (3.1) is referred to as softmax. In this expression, \( T \) denotes a temperature parameter and \( u_{i,k} = x_i^T a_{i,k} \) is the weighted sum of inputs \( x_i \) applied to the \( i \)th expert network. The values of the synaptic weight vector \( a_{i,k} \) of the \( i \)th output neuron in the gating network at iteration \( k+1 \) are adapted according to

\[
a_{i,k+1} = a_{i,k} + \eta \left( h_{i,k} - g_{i,k} \right) x_i,
\]

where \( \eta \) is a learning rate parameter, exponentially decreasing during time and converging to a minimum positive value.

The output of the \( i \)th expert network is associated with a conditional a posteriori probability, iteratively adapted with

\[
h_{i,k} = \sum_{j=1}^{K} g_{i,k} \frac{1}{\sigma_j} |x_j - y_{i,k} - d_i|^2 \sum_{j=1}^{K} g_{i,k} \frac{1}{\sigma_j} |x_j - y_{i,k} - d_i|^{-1},
\]

where \( d_i \) is the desired response, and \( \sigma_j \) denotes a scaling parameter associated with the \( i \)th expert network.

### 3.2 Mixture of Adaptive Kalman Filters

We propose to compose this modular framework with a number of AKFs running in parallel, each modeled with a different realization of the unknown parameters. In general, the unknown parameters can include different process noise models as well as different measurement noise models. In many cases the uncertainties in modeling the process and measurement are accounted for via the process and measurement noise. A recursive weighting function (i.e., the learning function of the gating network) is used to weight the outputs of the filters. The gating network “learns” which filter is performing best by examining a given performance measure.

The weighting information is also used to update the filters’ noise properties of each AKF using expressions (3.4) and (3.5). The covariance matrices \( Q_{i,k} \) and \( R_{i,k} \) of the \( i \)th AKFs are supposed to be diagonal matrices. The original algorithm of the mixture of experts is thus modified, allowing to update the diagonal elements of these matrices. These parameters are adapted according to:
\[ Q_{i,k+1} = Q_{i,k} (1 - \beta \eta h_i) , \quad \text{and} \]
\[ R_{i,k+1} = R_{i,k} (1 - \beta' \eta h_i) , \]
with \( \beta > 0 \) and \( \beta' > 0 \) two constant learning coefficients.

To improve the learning performance of the gating network, an adaptive learning rate scheme was implemented: \( \eta = \eta (1 - e^{-\alpha t}) \), where \( \alpha \) is a positive constant, and \( \eta \) an exponentially decreasing function of time converging to a minimum positive value. This rate is not only a function of time but also of the last prediction error:
\[ \varepsilon = (d_i - y_{i,k-1})^T (d_i - y_{i,k-1}) . \]

Vectors \( d_i \) and \( y_{i,k} \) now represent respectively the last measurement and the last prediction. The purpose of this adaptive learning rate is to emphasize the current expert network adaptation. Each AKF, by being specialized very rapidly in only a part of the system's workspace, will thus improve its prediction performances.

The different realizations of AKF are competing together and the learning network mediates their outputs. The modular adaptive Kalman filtering approach introduced in this paper has certain key advantages over conventional filter banks. The gating network adapts the weights of individual filter outputs based on the observed sequence of measurement. This learning is unsupervised and the desired parameters need not to be specified.

A similar modular framework has been proposed with AKFs that are only adaptive in the sense of the noise statistics, using only realizations of the different possible noise statistics models (covariance matrices \( Q_i \) and \( R_i \)). With state model AKFs, the mixture of experts structure is not confronted to the system model derivation problem as with conventional KFs. Online update rules enable the mixture of AKFs to cope with time varying environments.

4 Application to Robotic Visual Servoing

Vision plays an important role in robotics, and is referred to as visual servoing when introduced in a control feedback loop. One of the basic tasks of such a system is to position the robot end-effector at a desired location in the visual feature space. In the considered application, a three-degree-of-freedom redundant robot manipulator has to track a moving object in a 3-D environment. Due to the image acquiring process, the manipulator is controlled by data delayed of one or two image sampling periods. Visual servoing approaches often suffer from this time-delay problem. We propose to insert a prediction step in the visual control loop, where predicted values replace the actual image features.

This idea has already been applied to visual servoing. For example, the effectiveness of a prediction step has been demonstrated on a pickup task of a moving object. This application was based on the Kalman predicting approach using camera and object motion models. However, only the modeling and control of specific movements were considered (e.g., linear and circular motions, or constant velocity motions). Compared to this approach, ours requires more computational resources, but will be able to handle every kind of trajectories including trajectories with time-changing parameters.

4.1 Control Algorithm: the Jacobian Estimation

Visual servoing techniques fall into two categories: position-based and image-based. In position-based control, visual information is interpreted and transformed into information
with respect to a world coordinate system. This is a model-based approach that requires a precise location of the target in world coordinates, implying tedious robot and cameras calibration. Position-based visual servoing is therefore unsuitable for a model-independent approach.

Image-based visual servo control acts on an error signal which is defined by image features. This error is calculated by comparing the target position with the end-effector position. Most feature-based control approaches use the product of the image and robot Jacobians. These Jacobians relate differential changes in joint angles to differential changes in image features and their inverses are used to control the robot system. Jacobians can be determined by various methods: analytical derivation or non-parametric online estimation.

Online Jacobian estimation does not rely on a priori knowledge, e.g., camera model, kinematic robot model, and system parameters. Several researchers have already addressed the issue of adaptive camera-manipulator coordination. Generally, they estimate the inverse kinematics of the system by evaluating its performances for online real experiments, taking the unknown factors affecting modeling and prediction into account.

Our approach consists in estimating the inverse kinematic transformation between image and joint position spaces with a supervised self-organizing map (SOM). The visual sensor allows to measure the error between the target and the end-effector locations in the image feature parameter space $V \in \mathbb{R}^3$:

$$ e_k = d_k - v_k, \quad (4.1) $$

where $v_i \in V$ and $d_i \in V$ are respectively the image coordinate vector of the end-effector and of the visual target, and where $k$ is the time index.

Consider $f$, the forward kinematic transformation between joint and image position spaces:

$$ f: Q \rightarrow V, \quad v_k = f(q_k), \quad (4.2) $$

where $Q \in \mathbb{R}^q$ is the robot joint space of dimension $q$ and $q_k$ the joint angles vector. Feedback control methods are generally based on a linear approximation in the neighborhood of the working point. Thus, if $\partial q_k$ and $\partial v_k$ are small displacements measured respectively in joint and image feature spaces at instant $k$, a local linear Jacobian approximation determines the motor response $\Delta q_k$ in terms of an incremental position variation:

$$ \Delta q_k = f^{-1}(q_k) e_k = \hat{F}_k e_k, \quad (4.3) $$

The inverse image Jacobian $\hat{F}_k = f^{-1}(q_k)$ is approximated with a supervised SOM neural network as proposed and fully described in.

If $M$ is the lattice of the map (in our application $M$ is a grid of 15 x 15 x 20 neurons), each neuron of the map is associated to output synaptic weights $w_{ij}^{(out)}$. At time $k$, $\hat{F}_k$ is approximated by the best selected weights: $\hat{F}_k = w_{i(q)k}^{(out)}$, where $i(q) \in M$ is the best-matching (winning) neuron. This neuron $i(q)$ is found using the minimum-distance Euclidean criterion:

$$ i(q) = \text{argmin}_j \|q_k - w_{j}^{(in)}\|, \quad j \in M. \quad (4.4) $$

Defining $\Lambda_{i(q)k}$ as a neighborhood function centered around the winning neuron $i(q)$, $\mu_k$ and $\lambda_k$ as time-varying learning rates, the synaptic weights of all neurons are adjusted according to:
\[ w^{(\text{in})}_{j,k+1} = \begin{cases} w^{(\text{in})}_{j,k} + \mu_k (q_j - w^{(\text{in})}_{j,k}) & \text{if } j \in \Lambda w_{i(q_k),k}, \\ w^{(\text{in})}_{j,k} & \text{otherwise} \end{cases}, \quad (4.5) \]

\[ w^{(\text{out})}_{j,k+1} = w^{(\text{out})}_{j,k} + \lambda_k \frac{(\Delta q_k - e_k^T w^{(\text{out})}_{j,k} e_k)}{\|e_k\|^2}, \quad j \in \Lambda w_{i(q_k),k}, \quad (4.6) \]

This iterative stochastic adaptation of \( \hat{F}_k \) realizes de facto a linear local approximation and is used to control the robot arm. This approximation allows a model-independent control of the robot and overcomes the laborious calibration phase. Its performances have been emphasized in several robot positioning tasks and in sensory-motor coordination learning.

For the experiments presented in this paper, the supervised SOM that estimates the inverse Jacobian relationship has been trained in an earlier phase. The approximation is supposed to be sufficient to allow a precise positioning.

### 4.2 The experimental setup

In the experiments, the robot’s base is located at the origin of the 3-D world coordinate system. The scene is perceived by a fixed stereoscopic visual sensor placed at a distance of 1.5 m in the \( x \) direction (see Fig. 3). The resolution of each camera is 640 x 480 pixels with a focal of 50 mm. The three segments of the arm have a length of \( l_1 = 130 \), \( l_2 = 310 \), and \( l_3 = 330 \) mm. The robot dynamics is not a restrictive factor. Its maximum speed is of 10°/s per axes, viz, 1.6 m/s on the boundaries of the robot’s workspace.

A simple video-rate image processing provides the end-effector and the target positions in the image space. The proposed mixture of AKFs is then used to predict the future position and velocity of the target and the end-effector.

To meet the real-time implementation constraint, image processing is based on a windowing technique and takes advantages of the predicted features to avoid window error locations. The reliability of the whole control loop is thus increased.

Various experiments were conducted to evaluate the performances of the prediction step on adaptive visual servoing tasks. Some experiments have been implemented on our robotic platform. Other experiments use a simulation of the robotic setup to test the various system parameters in a more exhaustive manner.

### 4.3 A First Experience on a 3-D Linear Motion

To demonstrate the prediction and target tracking performances of the mixture of experts, a preliminary experiment is conducted on a simple linear target moving with constant velocity to ensure time-invariant parameters.

Simulations were run for a target moving linearly from location \([200, -20, 90]\) to location \([340, 10, 150]\) at a constant velocity of 0.162 m/s. For the simulations, 8 AKFs are made potentially available to predict the target movements.

The simulation results show that one AKF is always selected to represent the trajectory, and in some cases a second one is also selected covering the transient response (about the 15 first iterations). The selection algorithm only retains a single AKF because the parameters are time-invariant.

Fig. 2 shows the path of the target in both the left and right images as well as the path followed by the robot end-effector. A single AKF, with adequate noise properties, is able to predict this motion with a good precision and permits a successful object tracking.

This is only possible if the target’s dynamics or a priori knowledge is available.
4.4 Tracking Results on a Complex 3-D Unknown Motion

The problem addressed in this study is the one of controlling the end-effector of a robotic manipulator arm on a visual target with unknown time-varying movements. The application is done with an accelerating curved trajectory (a 17 order polynomial) which ends with little random movements in a restricted area. The speed of the target varies between 0.01 and 1.5 m/s.

The predictive control is implemented with a mixture of experts architecture composed of 40 AKFs with different parameters and initializations. Fig. 4 shows the weightings of the prediction for the successive AKFs. For unchanging system parameters, the gating network favors the most suitable AKF to do the predictions. The selected AKFs are alternatively reinforced to predict specific kinds of trajectories. Fig. 3 also shows the target trajectory and the end-effector position evolution in the 3-D space.

Comparisons have been realized to demonstrate the performance of the learning algorithm and the AKF assignment. The ANN of the gating network is compared to a gating network with short term memory (which gives a weight $g_{i,k}$ of 1 to the last best AKF and of 0 to all others) and to a gating network which is a random process. The results are presented in Table 1 and are compared to a single AKF working alone. The examination of the results leads to different remarks. The gating network adapted with the above-mentioned gradient ascent procedure is the most successful technique because...
it performs an efficient association between AKF performances and dynamics. The selected AKFs can thus be alternatively specialized in an efficient manner. The mixture of AKFs, working with a short term memory gating network, switches between the different dynamic models too rapidly, without choosing the adequate one. Due to its lack of learning capabilities, this gating network is not able to allow for the evolution of the inputs, cannot discern the innovation and thus is unable to assign a new AKF if necessary. The mixture with a random process as gating network does not allow learning either. The AKFs are not associated to the target’s dynamics, and thus cannot properly adapt their parameters to increase their prediction performances. The single AKF is limited: its structure is not suited and designed for different models of dynamics and target behaviors. Indeed, its noise properties are not valid over the whole trajectory, inducing important maximum errors.

The different experiments prove that the mixture of state model AKFs is able to predict every trajectory, even those of targets with changing properties. To cover the wide part of the robot arm’s workspace, the mixture of state model AKFs, with different realizations of the unknown parameters and with its learning properties, is an elegant solution to fit the target’s model and is thus accurate for all target properties. Once the object position is predicted, the robot can be controlled by conventional image-based visual servoing. For the present experiment, the neuro-controller associated with the modular predictor allows to divide the tracking error by 3 resulting in an error of 9.15 mm (max. 46.25) in the 3-D space. It should be emphasized that the error is essentially along the direction of the cameras focal line (x-axis). With the modular prediction step, the tracking error is 8.58 mm (max. 44.96) in the x-axis while it amounts to 1.65 mm (max. 20.62) and 1.59 mm (max. 17.13) respectively in the y and z dimensions.

Further experimentations have been done trying to evaluate the number of experts which is appropriate for a given complexity. It can be deduced that a mixture must always include more experts networks than necessary. If the system dynamics changes, the switching between two AKFs must arise, allowing for the specialization in regions. In the last application, about 20 of the 40 AKFs were selected to be representative of a region.
The others never participate to the prediction task or only slightly during transition phases. They have not been associated to a region of the input space, and stay ready for new movements. The AKFs that are not selected do not consume any computational resources. Raising indefinitely the number of experts networks will not increase the mixture’s performances, but a minimum number of expert networks is required for a good specialization and therefore for good performances. A compromise must be found between performances and computational costs. The fact that the filters are adaptive permits to considerably reduce the number of expert networks; in predicting targets movements, 40 AKFs allow to predict very complex trajectories.

### 5 Conclusion

The Kalman filter assumes that a process’ dynamics can be modeled, and that noises that affect the state of the system and the sensor data are stationary and zero-mean. While this filter is an optimal estimation technique that is well-suited for many different problems, its limitations are the properties of the noises and the system’s changing parameters and nonlinearities. To overcome these problems, we propose state model adaptive Kalman filtering schemes, organized in a modular learning framework known as mixture of experts and based on the divide-and-conquer principle. The end result is a modular framework consisting of isolated clusters of adaptive Kalman filters whose individual estimations are mediated by a gating network.

In this structure, each filter uses an adaptive state model which fits the system’s current state and uses different noise properties. The predictions of all adaptive Kalman filters participate to the global prediction and a learning scheme arbitrates among the different expert networks in an unsupervised fashion, by assigning higher values to adaptive filters that are expected to be close to the desired response for a given input value.

This modular adaptive prediction scheme, through the state model adaptation and through noises properties adaptation, is an elegant solution for time-varying environments and allows to use a limited and moderate number of filters to cover all possible realizations.

The performances of this mixture of adaptive Kalman filters are evaluated by a prediction task for a visual servoing problem: the pursuit of a moving object in a 3-D space with a 3 degree-of-freedom robot manipulator. Inserted in an adaptive control loop, the mixture of filters predicts the target and the end-effector’s future positions to compensate the time-delay of the visual processing. The manipulator is efficiently controlled with predicted image coordinates. The expert networks and the gating network are all trained online and simultaneously. In this application, the adaptive filters use a transition matrix which is not shown here.

#### Table 1. The absolute prediction errors (in pixels) for a single AKF and mixtures of experts with the softmax learning algorithm, with short term memory and with a random weighting.

<table>
<thead>
<tr>
<th>Mixture of AKFs</th>
<th>X coordinate</th>
<th></th>
<th>Y coordinate</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Mean</td>
<td>Std Dev.</td>
<td>Max.</td>
<td>Mean</td>
</tr>
<tr>
<td>With softmax gating network</td>
<td>0.37</td>
<td>2.72</td>
<td>22.11</td>
<td>0.29</td>
</tr>
<tr>
<td>With short term memory gating</td>
<td>1.02</td>
<td>11.82</td>
<td>144.46</td>
<td>1.02</td>
</tr>
<tr>
<td>network</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>With random gating network</td>
<td>31.27</td>
<td>70.30</td>
<td>148.36</td>
<td>23.35</td>
</tr>
<tr>
<td>Single AKF</td>
<td>0.52</td>
<td>9.43</td>
<td>178.48</td>
<td>0.43</td>
</tr>
</tbody>
</table>
continuously representative of the system’s current state value, and we use the mixture of expert concept to detect, estimate and compensate abrupt changes in the target motion and every variation of system parameters. We have shown that the robot can adapt itself to uncertain environments by using the mixture of AKFs even though the object’s speed and motion are unknown.

References