Controlling Model Complexity in Flow Estimation

Z. Duric, F. Li, H. Wechsler
Department of Computer Science
George Mason University
Fairfax, VA 22030
{zduric,fli,wechsler}@cs.gmu.edu

V. Cherkassky
Department of Electrical Engineering
University of Minnesota
Minneapolis, MN 55455
cherkass@ece.umn.edu

Abstract

This paper describes a novel application of Statistical Learning Theory (SLT) to control model complexity in flow estimation. SLT provides analytical generalization bounds suitable for practical model selection from small and noisy data sets of image measurements (normal flow). The method addresses the aperture problem by using the penalized risk (ridge regression). We demonstrate an application of this method on both synthetic and real image sequences and use it for motion interpolation and extrapolation. Our experimental results show that our approach compares favorably against alternative model selection methods such as the Akaike's final prediction error, Schwartz's criterion, Generalized cross-validation, and Shibata's model selector.

1. Introduction

Statistical Learning Theory (SLT) [13, 5] provides the mathematical framework for estimating motion models from finite training data; it enables a better understanding of problems related to generalization and facilitates the development of rigorous learning algorithms; it provides analytical generalization bounds for model selection, which relate unknown prediction risk (generalization performance), and known quantities such as the number of training samples, empirical error, and a measure of model complexity called the Vapnik-Chervonenkis (VC) dimension. In this setting, model selection amounts to *model complexity control*.

Many computer vision problems, including motion analysis, registration, segmentation, and stereo, require optimal estimation using regression. Motion estimation from image sequences is "a difficult problem that involves pooling noisy measurements to make reliable estimates", it "assumes some model of the variation within the region" [3]. The goal of this paper is to choose, from a small noisy data set, an optimal model that would yield minimum error for unseen inputs (i.e., minimum prediction risk). Model selection criteria are used to select a *correct* motion model from several possible motion models; its assumed that one of the available models is the true motion model. This setting is much simpler than the general problem of model selection [5], where the set of possible models may not contain the true model.

This paper describes an application of an SLT-based model selection to the problem of estimating optimal motion models from small sets of image measurements (normal flow). We address the aperture problem using the SLT formalism of penalized risk (ridge regression). We present the results of applying the estimated motion models to motion interpolation and extrapolation on both synthetic and real image sequences for both motion interpolation and extrapolation; these results demonstrate the feasibility and strength of our approach. Experiments on synthetic data show that our approach compares favorably against alternative model selection methods, such as the Akaike's final prediction error (*fpe*), Schwartz's criterion (*sc*), Generalized cross-validation (*gcv*), and Shibata's model selector (*sms*).

Section 2 briefly reviews model selection in predictive learning and VC-based analytic methods for model selection; it introduces the SLT-based model selection criterion used in this paper. Section 3 describes our approach. It discusses motion estimation and describes the application of the SLT to motion model selection. Section 4 reviews the experimental results on a synthetic and a real image sequence. Finally, Section 5 presents the conclusions.

2. Model Selection for Regression

A learning method is an algorithm that estimates an unknown mapping (dependency) between system's inputs and outputs from the available data—i.e., known (input,output) samples. Once such a dependency has been estimated it can be used for prediction of system outputs from the input values. The usual goal of learning is the prediction accuracy, i.e. generalization.

In the regression formulation, the goal of learning is to estimate an unknown (target) function $g(\mathbf{x})$ in the relationship $y = g(\mathbf{x}) + \epsilon$, where the random error (noise) is zero mean, **x** is a d-dimensional vector and y is a scalar output. A learning method selects the *best* model $f(\mathbf{x}, \omega_0)$ from a set of possible models $f(\mathbf{x}, \omega)$ specified *a priori*. The quality

Proceedings of the Ninth IEEE International Conference on Computer Vision (ICCV 2003) 2-Volume Set
0-7695-1950-4/03 $17.00 © 2003 **IEEE**

of an approximation is measured by the loss (discrepancy) measure $L(y, f(\mathbf{x}, \omega))$. A common loss function for regression is the squared error. Thus learning is the problem of finding the function $f(\mathbf{x}, \omega_0)$ that minimizes the prediction risk functional

$$R(\omega) = \int (y - f(\mathbf{x}, \omega))^2 p(\mathbf{x}, y) d\mathbf{x} dy$$

using only the training data $(\mathbf{x}_i, y_i), i = 1, \ldots, n$, generated according to some unknown joint probability density function (pdf) $p(\mathbf{x}, y) = p(\mathbf{x})p(y|\mathbf{x})$. Prediction risk functional measures the accuracy of the learning method's predictions of the unknown target function $g(\mathbf{x})$.

The standard formulation of the learning problem defined above amounts to function estimation from a set of admissible functions. Here the best function (model) is the one minimizing the prediction risk. The problem is ill-posed since the prediction risk functional is unknown (by definition). Most learning methods implement the idea known as *empirical risk minimization* (ERM), which is choosing the model minimizing the empirical risk, or the average loss for the training data:

$$R_{emp}(\omega) = \frac{1}{n} \sum_{k=1}^{n} (y_k - f(\mathbf{x}_k, \omega))^2. \qquad (1)$$

The ERM approach is only appropriate when the parametric form of unknown dependency is known. In such a parametric approach the unknown dependency is assumed to belong to a narrow class of functions specified by a given parametric form. In most practical applications parametric assumptions do not hold true and the unknown dependency is estimated in a wide class of possible models of varying complexity.

Since the goal of learning is to obtain a model providing minimal prediction risk, it is achieved by choosing a model of *optimal complexity* corresponding to smallest prediction (generalization) error for future data. Existing provisions for model complexity control include [5] penalization (regularization), weight decay (in neural networks), parameter (weight) initialization (in neural network training), and various greedy procedures (a.k.a. constructive, growing or pruning methods). Classical methods for model selection are based on asymptotic results for linear models. Recent approaches based on approximation theory extend classical rate-of-convergence results to nonlinear models (e.g. multilayer perceptrons); they are, however, still based on asymptotic assumptions. Non-asymptotic guaranteed bounds on the prediction risk for finite-sample settings have been proposed in VC-theory [13].

There are two general approaches for estimating prediction risk for regression problems with finite data: analytical and data-driven. Analytical methods use analytic estimates of the prediction risk as a function of the empirical risk penalized by some measure of model complexity. Once an accurate estimate of the prediction risk is found, it can be used for model selection by choosing the model complexity that minimizes the estimated prediction risk. In the statistical literature, various analytic prediction risk estimates have been proposed for model selection (for linear regression). These estimates take the form of:

$$R_{est}(\omega) = r\left(\frac{d}{n}\right) \frac{1}{n} \sum_{k=1}^{n} (y_k - f(\mathbf{x}_k, \omega))^2, \qquad (2)$$

where r is a monotonically increasing function, d is the model complexity (number of degrees of freedom), and n is the training sample size. r is often called a penalization factor because it inflates the average residual sum of squares for increasingly complex models.

SLT provides analytic upper bounds on the prediction risk that can be used for model selection [13]. To make practical use of such bounds for model selection, one has to choose the practical values for theoretical constants involved [5, 6]; this gives the penalization factor, $r(p, n)$, called the Vapnik's measure (vm):

$$r(p, n) = \left(1 - \sqrt{p - p \ln p + \frac{\ln n}{2n}}\right)_+^{-1} \qquad (3)$$

where $p = h/n$, h denotes the VC-dimension of a model and $(\cdot)_+ = 0$, for $x < 0$. In SLT $R_{est}(\omega)$ is obtained by substituting $r(p, n)$ for $r(d/n)$ in Eq. (2). For linear estimators with m degrees of freedom, the VC-dimension is $h = m$. The model providing minimal prediction risk $R_{est}(\omega) = r(p, n)R_{emp}(\omega)$ (see Eqs. (1-3)) is then chosen.

3. Controlling Motion Model Complexity

The regression problem in motion estimation has a *training* set of examples consisting of image points $\mathbf{x}_n$ and measurements y_n, which put in correspondence image points in two or more consecutive frames frames from a video sequence. The goal of training is to learn how to model the dependency of the measurements on the inputs. The objective is to make accurate predictions on image points that were not included in the training set (*interpolation*), or on image points from unseen future frames (*extrapolation*). We will demonstrate our approach on an image sequence of a moving arm (see Fig. 1).

3.1. Normal Flow

Normal flow is the projection of image motion (optical flow) onto the edge gradient direction [2]. It is usually computed from image derivatives resulting in very noisy mea-

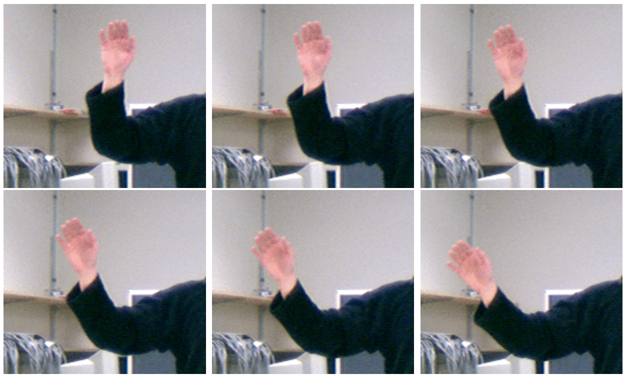

Figure 1: Frames 2,4,6,8,10, and 12 from a 13-frame sequence of a moving arm.

surements. In addition, since it corresponds to edge motion in the normal direction only it gives rise to the *aperture problem*—i.e., in a small region along a straight edge it does not contain any information about tangential motion of the edge. This problem is usually solved by assuming that the motion in a sufficiently large region, that includes edges of varying orientations, obeys some simple model so that the information over the whole region can be used to recover the missing information. The method used here estimates the normal flow from pairs of successive color image frames without image derivatives [8]. The normal flow computed from images in Fig. 1 is shown in Fig. 2. The SLT assumes that all outlier measurements have been rejected before parameter estimation. We use the folowing method that appears quite effective in rejecting most outliers. A connected component algorithm is applied to all points at which normal flow was computed. Two pixels with normal flows $\vec{u}_1, \vec{u}_2$ are connected if they are 8-neighbors and $\|\vec{u}_1 - \vec{u}_2\| \leq \epsilon$, where $\epsilon = \max\{c_1, c_2\|\vec{u}_1 + \vec{u}_2\|\}$. All connected components smaller than five pixels are removed as outliers. In the experiments presented in this paper $c_1 = 2$ and $c_2 = 0.2$ were used.

3.2. Motion Model Estimation

A hierarchy of parametric flow models has been proposed including pure translation, image rotation, 2D affine flow, and 2D homography (8-parameter or simplified quadratic flow). We will consider all those models here. 8-parameter flow corresponds to the instantaneous projected image motion field generated by a moving plane. Other models used here can be obtained by setting some of the eight parameters to zero. In the 8-parameter model coordinates of a point (x, y) in the first frame will move to (x', y') in the next frame:

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} w_1 \\ w_4 \end{pmatrix} + \begin{pmatrix} w_2 & w_3 \\ w_5 & w_6 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix}$$

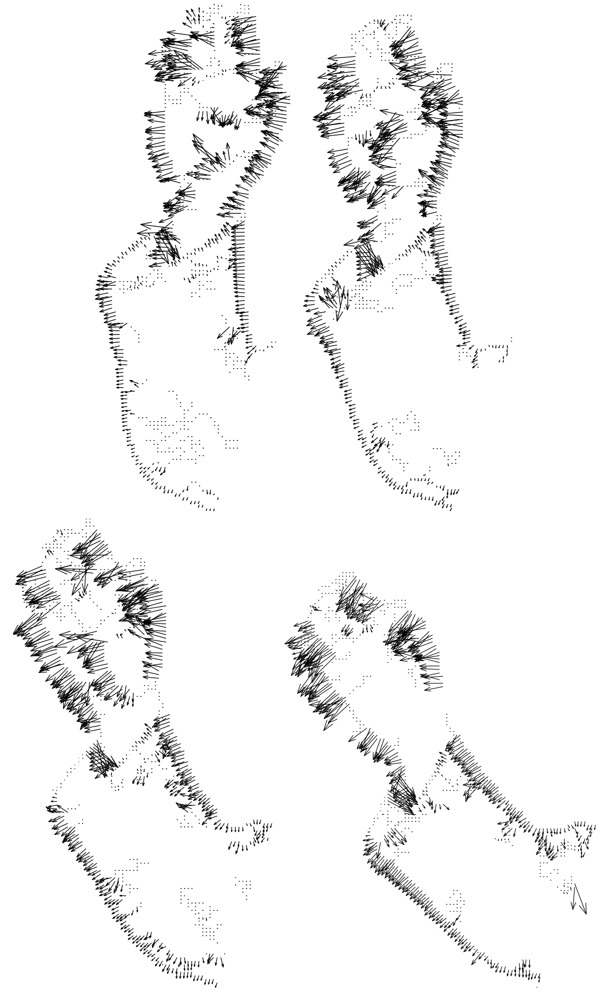

Figure 2: Normal flow computed from pairs of frames 2-3, 4-5, 7-8, and 10-11 of the moving arm sequence.

$$+ \begin{pmatrix} x^2 & xy \\ xy & y^2 \end{pmatrix} \begin{pmatrix} w_7 \\ w_8 \end{pmatrix} \quad (4)$$

Eq. (4) relates corresponding points in successive image frames. To obtain the displacement $\vec{u}(x, y) = (\delta x \;\; \delta y)^T$ of (x, y) we subtract $(x \;\; y)^T$ from both sides of (4). The left hand side of (4) is replaced by $(\delta x \;\; \delta y)^T$ and on the right hand side w_2 and w_6 get replaced by $w_2^1 = w_2 - 1$ and $w_6^1 = w_6 - 1$. The normal displacement field at (x, y) is given by $u_n(x, y) = \delta\vec{r}_n \cdot \vec{n} = n_x \delta x + n_y \delta y = w_1 n_x + w_2^1 x n_x + w_3 y n_x + w_7 x^2 n_x + w_8 xy n_x + w_4 n_y + w_5 x n_y + w_6^1 y n_y + w_7 xy n_y + w_8 y^2 n_y = \mathbf{w} \cdot \mathbf{p}$, where $\vec{n} = n_x \vec{\imath} + n_y \vec{\jmath}$ is the gradient direction, $\mathbf{p} = (n_x \;\; x n_x \;\; y n_x \;\; n_y \;\; x n_y \;\; y n_y \;\; x^2 n_x + xy n_y \;\; xy n_x + y^2 n_y)^T$, and $\mathbf{w} = (w_1 \;\; w_2^1 \;\; w_3 \;\; w_4 \;\; w_5 \;\; w_6^1 \;\; w_7 \;\; w_8)^T$ is the vector of affine parameters.

We use the method described in the previous section to compute the normal flow. For each edge point $\vec{r}_i$ we have one normal flow value $u_{n,i}$, that we use as an estimate of the normal displacement at the point, a vector $\mathbf{p}_i$ computed from (x_i, y_i) and $\vec{n}_i$, and an approximate equation $\mathbf{w} \cdot \mathbf{p}_i \approx u_{n,i}$. Let the number of edge points be $N \geq 8$. We need to find a solution of $P\mathbf{w} - \mathbf{b} = \mathbf{e}$, where **b** is an N-element vector with elements $u_{n,i}$, *P* is an $N \times 8$ parameter matrix with rows $\mathbf{p}_i$, and **e** is an N-element error vector. We seek the affine model **w** that minimizes $\|\mathbf{e}\| = \|\mathbf{b} - P\mathbf{w}\|$; the solution satisfies the system $P^T P w = P^T \mathbf{b}$ and corresponds to the linear least squares (LS) solution.

3.3. Model Selection

Training data consists of normal flow displacements. Affine and quadratic models are responsible for data generation. The motion estimation (learning) problem corresponds to choosing the best motion model from a given set of possible motions using the observed (training) data. The goal is to choose a model that will yield the lowest error at the image points not used at training. In this section we combine the SLT regression (see Sec. 2) and motion estimation techniques described in the preceding sections to choose a flow model that has the best predictive performance.

We use the square loss function—i.e., the squared difference $R_{est}(\mathbf{w}_m) = \frac{1}{n}\sum_{i=1}^{N}(u_{n,i} - u_{n,i}^m)^2$ between the computed normal flow $u_{n,i}$ and the predicted normal flow $u_{n,i}^m = \mathbf{w}_m \cdot \mathbf{p}_i$, where $\mathbf{w}_m$ corresponds to the estimated model. The task of model selection corresponds to choosing the best predictive model from a given set of linear parametric models, using a small set of noisy training data. We use VC-generalization bounds (3). The VC-dimension h (complexity) of a linear model is given by the number of degrees of freedom (DoF) of the model plus one.We choose from the following five models that we obtain by setting various elements of **w** to zero:

M_1: *pure translation*: $w_2^1 = w_3 = w_5 = w_6^1 = w_7 = w_8 = 0$, 2 DoF, $h = 3$;

M_2: *translation, shear, and rotation*: $w_2^1 = w_6^1 = w_7 = w_8 = 0$, 4 DoF, $h = 5$;

M_3: *translation and scaling*: $w_3 = w_5 = w_7 = w_8 = 0$; 4 DoF, $h = 5$;

M_4: *6-parameter affine*: $w_7 = w_8 = 0$. 6 DoF, $h = 7$;

M_5: *full affine, quadratic flow*: 8 DoF; $h = 9$.

3.4. Scaling and the Aperture Problem

In Section 3.2 we showed how to estimate the flow parameters **w** by solving the LS problem $\min \|P\mathbf{w} - \mathbf{b}\|$. Condition number $\kappa_2(P)$ is computed as the ratio of the largest and the smallest singular values of P:

$$\kappa_2(P) = \frac{\sigma_{\max}(P)}{\sigma_{\min}(P)}. \qquad (5)$$

The sensitivity in estimating **w** is roughly proportional to

$$\varepsilon(\kappa_2(P) + \rho_{LS}\kappa_2^2(P)) \qquad (6)$$

where ρ_{LS} is the magnitude of the residual of the LS solution and $\varepsilon = \|\Delta\mathbf{b}\|/\|\mathbf{b}\|$ is the relative error in **b**. The normal flow is computed with sub-pixel accuracy [8] and $\varepsilon = O(0.01)$. Since in the examples presented here $\rho_{LS} = O(1)$ it can be seen that condition numbers greater than 10 are undesirable. A large condition number typically corresponds to either inappropriate scaling of columns of P or to the aperture problem.

Scaling of columns P is handled as follows. The original problem is replaced by $\min\{\|(PG)\mathbf{y} - \mathbf{b}\|\}$. G is chosen to be a diagonal matrix whose elements are $\|P(:, i)\|^{-1}$, where $P(:, i)$ is the *i*th column of P. If the matrix PG is well conditioned, **y** is estimated using the LS method and $\mathbf{w} = G\mathbf{y}$ is computed. In the experiments with a moving forearm (see Figs. 1 and 2 and Sec. 4.3) typical values of $\kappa_2(P)$ are in the range 1 to 2 for M_1, in the range 30 to 40 for M_2 and M_3, in the range 40 to 50 for M_4, and in the range 3000 to 4000 for M_5. Scaling brings all these condition numbers below 5, i.e. $\kappa_2(PG) < 5$. This scaling procedure has been implemented and used for the experiments using real image sequences (see Sec. 4.3).

When the condition number after scaling, $\kappa_2(P)$, is still too large, it can be said that the data is not appropriate for the parameter estimation due to the aperture problem [2]. Note that the aperture problem refers to the fact that the flow cannot be estimated from the given normal flow due to the inappropriate distribution of feature points. This distribution is reflected in the data matrix P. In this case the solution is not provided by standard LS. The problem has to be solved by minimizing the penalized risk functional

$$R_{pen}(\mathbf{y}) = \frac{1}{n}(\|(PG)\mathbf{y} - \mathbf{b}\|^2 + \mathbf{y}^T\Phi\mathbf{y}) \qquad (7)$$

where Φ is a symmetric and nonnegative definite penalty matrix [5]. A reasonable choice of the penalty term is the ridge regression penalty function $\Phi = \lambda I$, where I is an identity matrix [9]. Solving the following modified least squares problem minimizes $R_{pen}(\mathbf{y})$ (Eq. (7)):

- Create the modified data matrices $U = \begin{pmatrix} PG \\ \sqrt{\lambda}I \end{pmatrix}$, $\mathbf{v} = \begin{pmatrix} \mathbf{p} \\ \mathbf{0} \end{pmatrix}$, where **0** is a column vector of zeroes.
- Minimize the empirical risk functional $R_{emp} = \frac{1}{n}\|U\mathbf{y} - \mathbf{v}\|$. The minimization is done by solving for **y** by LS method. Finally, compute $\mathbf{w} = G\mathbf{y}$.

- Compute the effective DoF for the penalized problem as $DoF = \sum_{i=1}^{m} \frac{\sigma_i^2}{\sigma_i^2+\lambda}$ where σ_i are the singular values of PG.

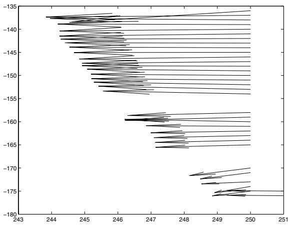

Figure 3: Normal flow vectors from a small region along the arm in Fig. 2.

λ is chosen to make $\kappa_2(P)$ small. For illustration purposes an experiment with $\kappa_2(U) = 10$ and $\kappa_2(U) = 100$ was performed for the example shown in Fig. 3. The data comes from a small region along the arm shown in Figs. 1 and 2. In this example the models M_1, M_2, and M_3 have small values of the condition numbers of the PG matrices. The condition numbers of data matrices PG for both M_4 and M_5, however, are very large ($>$ 1016) and their effective ranks are 5 and 7, respectively. The estimated empirical risk for models $M_1 - M_5$ are $(0.552, 0.015, 0.3, 0.0139617, 0.0131344)$ and the corresponding prediction risks are $(1.364, 0.049, 0.978, 0.059, 0.072)$. Note that based on Eqs. (2) and (3) M_2 would be chosen. However, if there were more feature points, either M_4 or M_5 could have been chosen since they have smaller empirical risk values than the other models; since they are poorly conditioned, however, penalized solutions have to be used. (Note that the penalization factor is heavy for small numbers of feature points.) The relevant results are as follows. In the case of the affine model M_4 the penalization coefficients $\lambda = (0.017, 0.17)$ result in condition numbers $(100, 10)$, empirical risk values $(0.0139622, 0.0190552)$, effective DoF values $(4.9969, 4.7258)$, and prediction risk values $(0.052, 0.068)$. The effective DoF are a crude estimate of the VC-dimension. It can be seen that the larger the penalization term more bias is introduced. In the case of the quadratic model (M_5) the penalization coefficients $\lambda = (0.020, 0.203)$ result in condition numbers $(100, 10)$ empirical risk values $(0.0131595, 0.022909)$, effective DoF values $(6.1726, 5.2896)$, and prediction risk values $(0.057, 0.088)$. Note that for small penalization factors the empirical risk goes up slightly, but the prediction risk goes down due to the lowered effective DoF.

4. Experimental Results

We performed experiments on both synthetic and real image sequences. In addition we compared our approach to four well-known model selection criteria.

4.1. Experiments on a Synthetic Image Sequence

A synthetic image sequence consisting of 11 frames of a 128-point moving square was generated using the following 6-parameter affine motion model:

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} 2.7 \\ -2.47 \end{pmatrix} + \begin{pmatrix} 0.991562 & 0.129631 \\ -0.129631 & 0.991562 \end{pmatrix}.$$

A noisy sequence was generated by corrupting all displaced by a Gaussian noise with a zero mean and variance 0.5. We performed both interpolation and extrapolation experiments on this sequence. In the interpolation experiment we randomly choose $n = 32, 64$ correspondences (out of 128) from pairs of successive frames and estimate the parameters for each of the motion models using LS (see Sect. 3.2). The bounds on prediction is computed for each model $M_1 - M_5$ using the LS error and its VC-dimension ($h = m + 1$). The corresponding interpolating total error is calculated using all the 128 points based upon ground truth information. The experiment is repeated 100 times with different random realizations of training data, for both non-noisy and noisy image sequences. The experimental results show that the prediction risk is a good model performance predictor for motion estimation for both non-noisy and noisy sequences. In addition, the prediction risk is a better predictor than the empirical risk. Ground truth M_4 is consistently found as the optimal motion model; its interpolation error is minimum. The quadratic model M_5 is a very close runner-up to the ground truth model M_4. In the extrapolation ("tracking") experiment we randomly subsample $n = 32, 64$ pixel correspondences from the first five frame pairs, and estimate the parameters for each of the models $M_1 - M_5$ using the LS method. Note that the data available for extrapolation is much less than the data available for the interpolation experiment. The prediction risk for each model is derived as before. The extrapolation error for each model for all the remaining frames (7-11) is calculated using the truth starting from frame 6. The experiment is repeated 300 times with different random realizations of training data for non-noisy and noisy image sequences. M_4 is chosen as the optimal model for motion tracking and its extrapolated error is minimum. The quadratic model M_5 is again a very close second to M_5.

4.2. Comparative Analysis of Model Selection Criteria

A comparative analysis of model selection criteria was performed for different forms of penalty r, which have been proposed in the statistical literature. They are listed below and compared to the Vapnik measure (*vm*), $p = h/n$.

- Final prediction error (*fpe*) [1]:

$$r(p) = (1 + p)(1 - p)^{-1}.$$

- Schwartz's criterion (*sc*) [12]:

$$r(p, n) = 1 + \frac{\ln n}{2} p(1 - p)^{-1}.$$

- Generalized cross-validation (*gcv*) [7]:

$$r(p) = (1 - p)^{-1}.$$

- Shibata's model selector (*sms*) [11]:

$$r(p) = 1 + 2p.$$

All these classical approaches are motivated by asymptotic arguments for linear models and therefore apply well for large training sets. In fact, for large n, prediction estimates provided by *fpe, gcv,* and *sms* are asymptotically equivalent. The penalization factor r inflates the average residual sum of squares for increasingly complex models. Our experimental results show that the Vapnik measure compares favorably against these criteria (see Table 1). The entries show the accuracy of predictions provided by each of the criteria for the synthetic image sequence.

Samples	*vm*	*fpe*	*gcv*	*sc*	*sms*
32	99.0%	91.0%	94.2%	89.3%	81.9%
64	96.8%	86.0%	87.0%	87.0%	82.6%
32	100%	98.2%	99.7%	98.3%	93.7%
64	100%	98.5%	99.0%	98.7%	96.7%
32	99.6%	91.3%	96.3%	90.8%	78.7%
64	99.5%	83.5%	88.1%	80.4%	74.4%
32	100%	95.6%	98.0%	96.0%	89.3%
64	100%	94.5%	96.3%	92.3%	88.0%

Table 1: Comparison of model selection criteria on the synthetic image sequence. Rows 1 and 2 show the interpolation results for the non-noisy sequence. Rows 3 and 4 show the extrapolation results for the non-noisy sequence. Rows 5 and 6 show the interpolation results for the noisy sequence. Rows 7 and 8 show the extrapolation results for the noisy sequence.

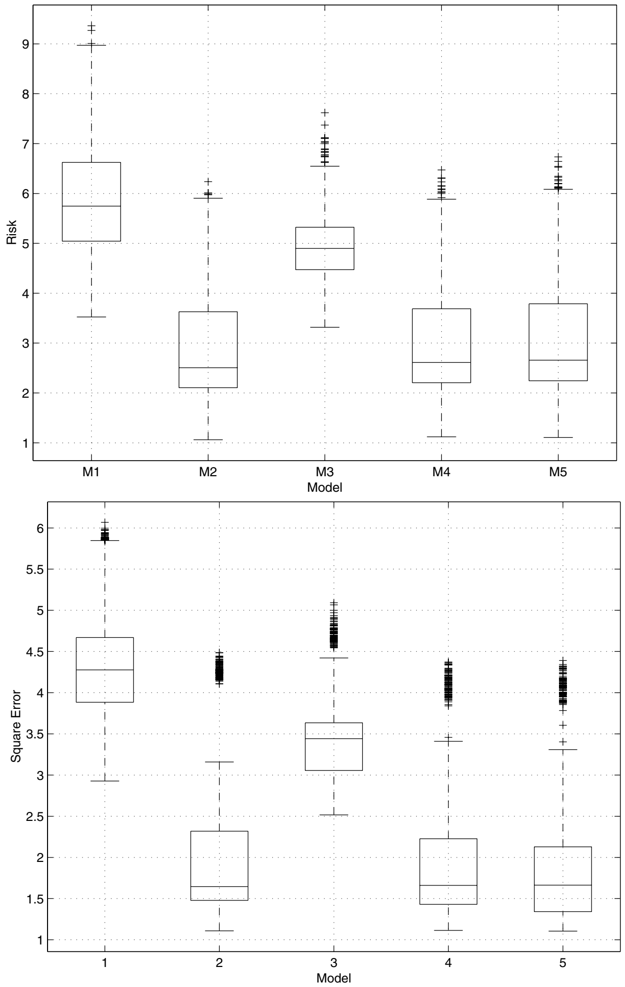

Figure 4: Interpolation results for the moving arm sequence. Top: prediction risk, bottom: square error.

4.3. Experimental Results for Real Image Sequence

Training data comes from eleven frames drawn from a real image sequence of a moving arm and the corresponding normal flow (see Figs. 1 and 2). We report the results obtained for both interpolation and extrapolation. The ground truth is not known and the images are inherently noisy. In the interpolation experiment we randomly subsampled 25% of image flow values (out of approximately 400 points); the experiment is repeated 100 times with different random realizations of training data. Training data is drawn from two frame pairs: $(i, i+1)$ and $(i+2, i+3)$; interpolation is performed for frames $(i+1, i+2)$. Fig 4 summarizes the prediction risk and interpolation error for the whole experiment. The prediction risk ranks the motion models so that its optimal choice, M_2, yields the minimum total interpolation error.

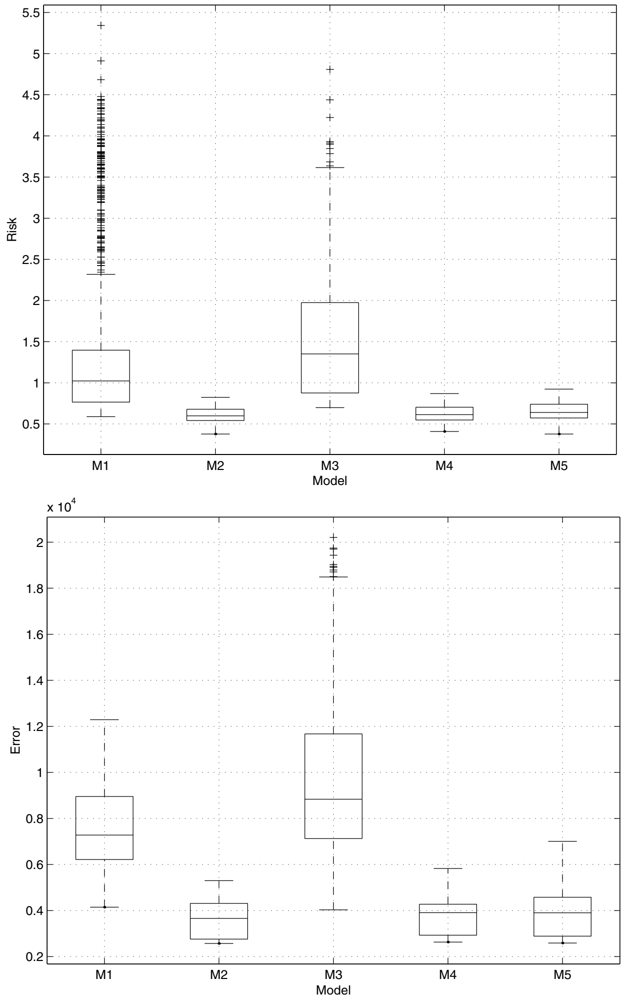

Figure 5: Extrapolation results for the moving arm sequence. Top: prediction risk, bottom: square error.

In the extrapolation experiment we randomly subsampled 10% of image flow values (out of approximately 400 points); the experiment is repeated 100 times with different random realizations of training data. Training data was drawn from the first five pairs of frames: $(i, i+1)$ starting with $i = 1$. Extrapolation is performed for the remaining five frames of the sequence. Fig. 5 summarizes the prediction risk and extrapolation error for the whole experiment. The prediction risk ranks the motion models so that its optimal choice, M_2, yields the minimum total extrapolation error.

5. Conclusions

This paper a novel application of Statistical Learning Theory (SLT) to optimal model selection, with applications to single motion estimation and tracking from small data sets of image measurements (flow). This is accomplished without using restrictive assumptions such as asymptotic settings and/or Gaussian noise. The experimental results, using both synthetic and real image sequences, demonstrate the feasibility and strengths of our approach for motion model selection using SLT. Our experimental results also show that our approach compares favorably against alternative model selection methods regarding the confidence they offer on motion estimation. The paper also shows how to address the aperture problem using the SLT formalism of (ridge regression) penalized risk.

References

[1] H. Akaike, “Statistical predictor information”, *Ann Inst. Of Stat. Math.*, 22:203–217, 1970.

[2] Y. Aloimonos and Z. Duric, “Estimating heading direction using normal flow”, *Int. Journal of Computer Vision*, 13:33–56, 1994.

[3] M.J. Black and P. Anandan, “The robust estimation of multiple motions: Parametric and piecewise-smooth flow fields”, *Computer Vision and Image Understanding*, 63:75–104, 1996.

[4] K. Bubna and C.V. Stewart, “Model selection techniques and merging rules for range data segmentation algorithms”, *Computer Vision and Image Understanding*, 80:215–245, 2000.

[5] V. Cherkassky and F. Mulier, *Learning from data*, Wiley, 1998.

[6] V. Cherkassky, X. Shao, F. Mulier, and V. Vapnik, “Model selection for regression using vc generalization bounds”, *IEEE Trans. on Neural Networks*, 10:1075-1089, 1999.

[7] P. Craven and G. Wahba, “Smoothing noisy data with spline functions”, *Numerische Math.*, 31:377-403, 1979.

[8] Z. Duric, F. Li, Y. Sun and H. Wechsler, “Using normal flow for detection and tracking of limbs in color images”, In *Proc. Int. Conf. on Pattern Recognition*, Quebec-City, Canada, 2002.

[9] G.H. Golub and C.F. Van Loan, *Matrix Computation*, 3rd. ed., John Hopkins University Press, Baltimore, MD, 1996.

[10] P.H.S. Torr, “An assessment of information criteria for model selection”, In *Proc. Computer Vision and Pattern Recognition*, 47–53, 1997.

[11] R. Shibata, “An optimal selection of regression variables”, *Biometrika*, 68:45-54, 1981.

[12] G. Schwartz, “Estimating the dimension of a model”, *Ann. Stat.*, 6:461–464, 1978.

[13] V.N. Vapnik. *The Nature of Statistical Learning Theory*, 2nd. Ed., Springer Verlag, 1999.