

David Nistér

Preemptive RANSAC for live structure and motion estimation

Received: 15 November 2004 / Accepted: 3 August 2005 / Published online: 25 November 2005
© Springer-Verlag 2005

Abstract A system capable of performing robust live ego-motion estimation for perspective cameras is presented. The system is powered by random sample consensus with preemptive scoring of the motion hypotheses. A general statement of the problem of efficient preemptive scoring is given. Then a theoretical investigation of preemptive scoring under a simple inlier–outlier model is performed. A practical preemption scheme is proposed and it is shown that the preemption is powerful enough to enable robust live structure and motion estimation.

Keywords Structure from motion · Real-time · Robust estimation · 3D-Reconstruction · Ego-motion

1 Introduction

This paper is about robust real-time estimation of the motion of a single camera in a mostly rigid scene. The estimation is performed with low delay as the motion occurs. Some structure of the scene is also estimated. Some examples are shown in Fig. 1. Structure from motion has been a highly active research area of computer vision for a long time and this paper owes a lot to a large number of contributions, more than we can hope to account for here. It is well known that estimates of structure and motion are highly sensitive to noise and outliers. It is therefore extremely important to minimize a cost function that corresponds as closely as possible to the true

noise and outlier conditions. It is widely accepted that once a good starting point is known, bundle adjustment, see e.g. [1], is the method of choice for achieving this. However, bundle adjustment and other polishing schemes will only converge to the right solution when given a good starting point. Thus, finding a good starting point under practical noise and outlier conditions is important.

Random sample consensus (RANSAC) [2] and similar hypothesize-and-test frameworks [3–6] have become the standard way of dealing with outliers arising from incorrect matches. Our system is based on this type of robust estimation and is in that respect a development in the direction taken e.g. by authors [7–12] and the commercial software (Boujou [13]). However, there are important differences. Our system operates in a calibrated framework where the intrinsic parameters are assumed known. This brings increased stability [14–16]. More importantly, it has the distinct advantage that the estimated camera motion and scene structure can be put in a metric coordinate frame directly, without delay due to self-calibration. Our system operates live and in real-time with low delay, which was not attempted by any of the authors cited earlier. Robust real-time estimation of a small number of degrees of freedom was reported by [17]. Our system does real-time estimation of general motion of a calibrated perspective camera. Another line of prominent work on real-time structure from motion is [18–20]. It is based on nonlinear filtering that fuses estimates over time. See [18] for a large number of additional references. We are going to focus on robust estimation, which we see as a way of feeding a nonlinear filter, or more generally a bundle adjuster.

In the RANSAC framework, motion hypotheses are generated from random minimal sets of observations. The hypotheses are scored based on their fit to the full set of observations and the best hypothesis is kept as the winner. We take the view of [3] that the scoring function should preferably be a robust likelihood or Bayesian cost function rather than a simple inlier count.

Inspired by [19] we use preemptive scoring of the motion hypotheses generated by random sampling. The preemption

Prepared through collaborative participation in the Robotics Consortium sponsored by the U.S. Army Research Laboratory under the Collaborative Technology Alliance Program, Cooperative Agreement DAAD19-01-2-0012. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation thereon.

D. Nistér (✉)
Sarnoff Corporation, CN5300, Princeton, NJ 08530, USA
E-mail: dnister@cs.uky.edu

Present Address: Center for Visualization and Virtual Environments,
Computer Science Department, University of Kentucky, 1 Quality
Street, Suite 800, Lexington, KY 40507-1464, USA

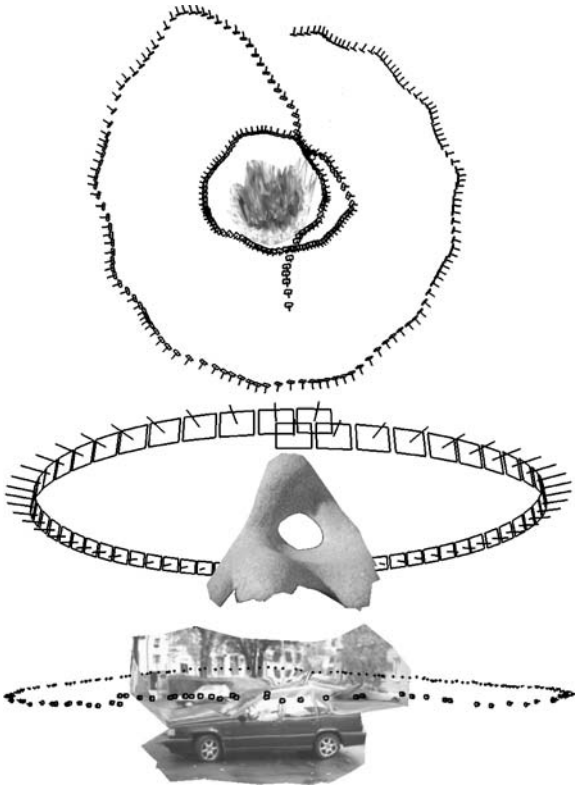


Fig. 1 Reconstructions made at real-time rates with low delay while loading the video source from disk. The top sequence was acquired by moving a hand-held camera around a flowerpot, first in a wide circle and then in a tighter circle, with some forward motion in between. The middle sequence is a turn-table sequence. No a priori knowledge of the motion was used. The bottom sequence was created by walking around a car with a hand-held camera

is used to avoid excessive scoring of useless motion hypotheses contaminated by outliers or distorted by noise. The problem is thus to find a scoring procedure that achieves high efficiency in a limited time. The preemptive scoring we propose uses a small fixed amount of time to choose between a predetermined number of hypotheses. It could be argued that this does not achieve a fixed confidence in the solution. However, in real-time applications the computation time budget has to be met and the goal is to achieve the highest possible confidence within the time limit.

All RANSAC schemes must by necessity be preemptive in some fashion since it is inconceivable to consider all possible minimal sets. The standard scheme employs a simple stop-criterion and all the motion hypotheses generated are exhaustively scored against all observations until the stop-criterion is satisfied. It has often been observed, see e.g. [9], that the standard stop-criterion is optimistic. This was clearly shown in [20], where guided sampling was used in the hypothesis generation to improve the situation. We concentrate on preemptive scoring, which does not put any constraints on the hypothesis generation. The rest of the paper is organized as follows. Section 2 discusses the relation to previous work on preemptive scoring. Section 3 outlines

robust estimation of camera motion in our calibrated framework. Section 4 introduces notation, definitions, and assumptions regarding preemptive scoring. The proposed preemption scheme is given in Sect. 5. Section 6 is a theoretical analysis of preemption in a simple inlier–outlier model. Section 7 gives results and Sect. 8 concludes.

2 Relation to previous work

The first preemption scheme of a non-trivial nature was proposed by [19]. A class of pretest called $T_{d,d}$ test is used to decide if a hypothesis should be dropped or exhaustively scored. For some small number d , the test is that all d -out-of- d randomly chosen observations should fit the hypothesis for the scoring to proceed. An approximate relation is derived for the optimal setting of d . Both standard RANSAC and the scheme of Chum can be implemented in what we refer to as depth-first manner, meaning that all scoring for a particular generated hypothesis is performed before moving on to the next hypothesis. We find another style of preemption more appealing. We call this style breadth-first preemption. We fix the number of hypotheses a priori, which lets us generate all our motion hypotheses first and then compare them on an equal footing. The appeal is that hypotheses can be measured against each other rather than against some absolute quality metric. A practical breadth-first preemption scheme with low overhead will be proposed. We will show that this preemption scheme is powerful enough to enable structure and motion estimation in real-time. We will use a simple inlier–outlier model and determine the most efficient preemption schemes from a certain subclass of preemption schemes in this model. We will also use the inlier–outlier model to indicate that the $T_{d,d}$ test is often too pessimistic about fully scoring hypotheses. Wasting perfectly good motion hypotheses is particularly harmful to efficiency when hypothesis generation is costly. Therefore, it is valuable to consider a more general $T_{c,d}$ test, where $c \leq d$. We will also indicate that in the simple inlier–outlier model, the most efficient preemption scheme always tries, in loose terms, to score the least previously scored observations and hypotheses.

3 The Robust calibrated framework

In the RANSAC paradigm, a number of random samples consisting of minimal sets are taken from the observations.¹ A minimal set contains the smallest number of observations needed to solve for the motion model. We refer to the algebraic solution that goes from the minimal set to the motion model as a *hypothesis generator*. In our calibrated framework, the five-point method [15] is used for two and three views. Pose with respect to known structure is solved for

¹ Schemes using non-minimal or less than minimal sets are also popular and the size chosen for the samples should perhaps be regarded as an integral part of the design of an efficient algorithm.

with the three-point method [21]. The five- and three-point methods yield multiple solutions and we typically use an additional point to select between the hypotheses, effectively creating 5-with-6 and 3-with-4 point methods, which is also a simple form of preemption. Coordinate frames are stitched together with a simple one-point method which is the calibrated analogue of the frame stitcher described in [9].

The hypotheses are scored according to their posterior likelihood and a unique hypothesis is singled out as the winner. For simplicity we will consider maximum likelihood, i.e. the case of a uniform prior. For the most part, the extension to an arbitrary prior is straightforward. We will ignore constant factors of the likelihood wherever possible. We use Cauchy distribution to model the errors, meaning that if the squared magnitude of a reprojection error is u , it contributes a term $-\ln(1 + u)$ to the log-likelihood. The \ln -function is expensive and the following trick saves it from becoming a major bottleneck. The reprojection errors are taken in groups of ten and the robust log-likelihood for each group is computed as

$$-\ln \prod_{i=1}^{10} (1 + u_i). \quad (1)$$

Larger groups should be avoided to ensure that the calculation stays within the floating point range. Each robust fit is finished off by iterative refinement to achieve the highest accuracy possible. In two views, we eliminate the structure parameters, but in three views full bundle adjustment is performed. The reason is that when fully optimized, bundle adjustment turns out to be just as fast as any attempts to eliminate the structure parameters.

4 Preemption preliminaries

A general theoretical analysis of preemption becomes quite complex. However, we will state the problem in its full generality in this section, before we make simplifying assumptions. Assume that we have a finite number of observations indexed by $o = 1, \dots, N$ and a sequence of hypotheses indexed by $h = 1, \dots, M$, which may also have an infinite range. Throughout the paper, we will assume that the observations have been randomly permuted to avoid any deterministic ordering (such as e.g. observations ordered from top to bottom of the image) that could otherwise ruin the quality of a preemptive scheme. Assume that we have a *scoring function* $\rho(o, h)$ that takes an observation index and a hypothesis index and gives a scalar value representing the log-likelihood of the observation given that the hypothesis is the correct motion model. The scalar value will be referred to simply as *term* since the log-likelihood $L(h)$ of the hypothesis indexed by h is

$$L(h) = \sum_{o=1}^N \rho(o, h). \quad (2)$$

We will occasionally use $L_i(h)$ to denote $\sum_{o=1}^i \rho(o, h)$. A *scoring sequence* is a sequence of index pairs $x_i = (o_i, h_i)$ indexed by i . An *order rule* ψ is a rule $x_i = \psi_i(x_1, \dots, x_{i-1}, \rho(x_1), \dots, \rho(x_{i-1}))$ that assigns the next pair in the scoring sequence given all the previous scores. A *preference rule* ϕ is a rule that selects the preferred hypothesis $\phi_i((x_1, \dots, x_i, \rho(x_1), \dots, \rho(x_i)))$ given all the scores so far. A *preemption scheme* $\Omega = (\psi, \phi)$ is defined by an order rule and a preference rule. The order rule defines in which order the scoring will occur and the preference rule determines which hypothesis is the best when the time budget is exhausted. We allow the preference rule to take on the value zero, $\phi_i = 0$ meaning simply that preemption is not allowed at i and that scoring has to proceed to the next valid preemption point. We will use Ω_i to denote the finite preemption scheme that scores up to i and then selects the preferred hypothesis ϕ_i . A preemption scheme is called *depth-first* if the scoring sequence obeys

$$h_i \leq h_j \quad \forall (i, j) : i \leq j. \quad (3)$$

A preemption scheme is called *breadth-first* if the scoring sequence obeys

$$o_i \leq o_j \quad \forall (i, j) : i \leq j. \quad (4)$$

A scheme that is neither depth-first nor breadth-first is called *hybrid*. Assume that the time spent evaluating one term $\rho(o, h)$ is a constant independent of o and h . Let this constant be the unit of time. Assume that the time spent generating a hypothesis is a constant τ . We will consider finite scoring sequences $X_i = (x_1, \dots, x_i)$. Let $|X|$ be the number of elements in X . Let the *hypothesis set* $H(X)$ be the set of hypotheses visited by X and $|H(X)|$ be the number of elements in $H(X)$. If we neglect overhead costs and assume that only the hypotheses in the hypothesis set are computed, the time $t(X)$ spent evaluating X is

$$t(X) = |H(X)|\tau + |X|. \quad (5)$$

A preemption scheme is called *lossless* if

$$L(\phi_i) \geq L(h) \quad \forall (i, h) : (\phi_i \neq 0), (h \in H(X_i)). \quad (6)$$

Thus, a preemption scheme is lossless if at a valid preemption point, no hypothesis from the hypothesis set can have a higher likelihood than the preferred hypothesis. A lossless scheme might make sense when the hypothesis generation cost is extremely high. On the other hand, under such circumstances the scoring is cheap compared to the hypothesis generation, so preemptive scoring is not very meaningful. Even the standard RANSAC scheme is a fundamentally stochastic algorithm without deterministic guarantees of finding the global maximum of the likelihood. We are not obligated to find the best hypothesis from some particular set of hypotheses, which have been generated randomly in the first place. The computational resources are always limited in practice and a lossy scheme can potentially consider more hypotheses than a lossless one. Therefore, a lossy

scheme can in fact be strictly better than a lossless scheme at maximizing the likelihood, which is our overall goal.

An *ideal lossless sequence* for a particular hypothesis set is the scoring sequence of a lossless preemption scheme with the smallest possible time among all lossless preemption schemes with that hypothesis set. An ideal lossless sequence for a particular set of observations and hypotheses is straightforward to compute and gives a performance bound for lossless schemes. As a simple rule of thumb, an ideal lossless sequence will be on the order of the number of outliers times $|H(X)|$ terms long (since that is the minimal number of scores required in order to prove that the best hypothesis is truly the best, if we assume that outliers cause a fixed penalty and inliers cause no penalty for the correct hypothesis), although this of course depends on the scoring function among other things.

We will regard the log-likelihood $L(\phi_i)$ of the preferred hypothesis as a random variable, where the randomness is caused by the random sampling and the random permutation of observations. We define the efficiency $E(\Omega_i)$ of a preemption scheme Ω_i to be the expected log-likelihood of the preferred hypothesis, i.e.

$$E(L(\phi_i)). \quad (7)$$

For fair comparison, the efficiencies of preemption schemes with the same computation time should be considered.

5 The proposed preemption scheme

The proposed preemption scheme is essentially breadth-first, although in practice we implement it as a hybrid to reduce overhead. An illustration is provided in Fig. 2. The scoring sequence is of a restricted type that is defined by a decreasing *preemption function* $f(i)$ $i = 1, \dots, N$ that indicates how many hypotheses are to be kept at each stage. The idea is to first generate all the hypotheses $h = 1, \dots, f(1)$ that will be used. All these hypotheses are then scored against the first observation. (Remember that we assume that the observations have been randomly permuted.) Based on the scores, we keep the best $f(2)$ hypotheses and score them against the

second observation. The process is repeated, scoring each observation o against the best $f(o)$ hypotheses so far. To summarize in algorithmic form:

- Algorithm 1**
1. Randomly permute the observations.
 2. Generate the hypotheses indexed by $h = 1, \dots, f(1)$.
 3. Compute the scores $L_1(h) = \rho(1, h)$ for $h = 1, \dots, f(1)$. Set $i = 2$.
 4. Reorder the hypotheses so that the range $h = 1, \dots, f(i)$ contains the best $f(i)$ remaining hypotheses according to $L_{i-1}(h)$.
 5. If $i > N$ or $f(i) = 1$, quit with the best remaining hypothesis as the preferred one. Otherwise, compute the scores $L_i(h) = \rho(i, h) + L_{i-1}(h)$ for $h = 1, \dots, f(i)$, increase i and go to Step 4.

The main reason why we find this approach appealing is because hypotheses can be compared against each other throughout the whole process, rather than against some absolute quality measure. One can approximate this in a depth-first scheme by comparing to previously scored hypotheses, but a significant amount of time can be wasted before the first good hypothesis is found. It could be argued that the breadth-first scheme is at the risk of wasting time on bad observations in the same way as a depth-first scheme is in danger of wasting time on bad hypotheses. However, note that in the typical RANSAC setting the hypotheses are composed from multiple observations and are typically useless if any one of those observations are outliers. Therefore, the fraction of good observations is much larger than the fraction of good hypotheses, which means that a naive breadth-first scheme will waste less of its time than a naive depth-first scheme. It is conceivable that it is possible to improve on the breadth-first scheme by starting with a diagonal scheme similar to the one discussed in Sect. 6 to pre-sort observations based on their relative merit. This may decrease the amount of time wasted on bad observations. However, it is not obvious what the effects of such a scheme will be. We find it comforting never to change the entirely random order of the observations. Ignoring overhead, the time spent in a preemption scheme of the type described by Algorithm 1 is

$$t(f) = f(1)\tau + \sum_{o=1}^N f(o). \quad (8)$$

In our real-time implementation, we use the preemption function

$$f(i) = \lfloor M2^{-\lfloor \frac{i}{B} \rfloor} \rfloor, \quad (9)$$

where $\lfloor \cdot \rfloor$ denotes downward truncation, and B is a block size. This gives an execution-time of approximately $M(\tau + 2B)$. The preemption function changes only after every B th observation, which means that the partial reordering Step 4 only has to take place every B th iteration. The quantile needed to perform the partial ordering of Step 4 is found in average linear time with the quickselect algorithm [24].

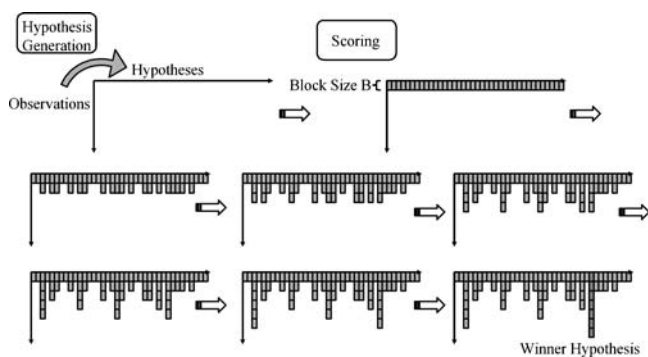


Fig. 2 An illustration of breadth-first preemptive scoring as proposed. The hypotheses are first generated using the observations. The hypotheses are then scored competitively in successive cuts, with the observations taken in a random but common order

6 Analysis of preemption in the inlier–outlier model

We will now turn to a theoretical study of preemption in a simple inlier–outlier model, which we will refer to as the inlier-and-model (*i.a.*-model). In a practical situation, the initial goal of the preemptive scoring is to quickly and efficiently weed out hypotheses that have been contaminated by outlier observations. The *i.a.*-model is useful for modeling this stage. Once only the noisy but otherwise good hypotheses are left, the goal of the scoring is to accurately discriminate between similar hypotheses using many observations. This stage is not modeled by the *i.a.*-model. Note however that our proposed preemption scheme can handle both stages. The iterative refinement used after the RANSAC process is also very efficient for achieving the goal of the second stage. Thus, the main responsibility of the preemptive scoring is to weed out contaminated hypotheses. See [25] for a variation on RANSAC geared at improving on the second stage.

In the *i.a.*-model, we make the following assumptions in addition to the ones made in Sect. 4:

1. An observation is either an inlier or an outlier.
2. A hypothesis is either an inlier or an outlier.
3. The scoring function $\rho(o, h)$ returns 1 if both o and h are inliers. Otherwise it returns 0.
4. Each hypothesis has a prior probability λ of being an inlier. Moreover, this prior probability is independent of the other hypotheses and the observations, ignoring e.g. the fact that in practice the hypotheses are generated from the observations.

Remember that N denotes the number of observations. Let n be the number of inlier observations. Then $\epsilon = \frac{n}{N}$ is the inlier fraction. We will use

$$\lambda = \epsilon^s, \quad (10)$$

where s is the sample size, i.e. the size of the minimal sets. In the *i.a.*-model, all uncontaminated hypotheses will have the same global likelihood. The efficiency is simply

$$E(\Omega) = nP, \quad (11)$$

where P is the probability of finding an inlier pair, i.e. the probability that the scoring sequence will contain some pair x_i with an inlier observation and an inlier hypothesis. In the sequel, we will call this a *good pairing*. We can now restrict our attention to studying P .

We first analyze the efficiency of preemption with the *c-out-of-d* test, i.e. $T_{c,d}$ test, in this model. In this model, outlier hypotheses never pass the test. Inlier hypotheses fail the test with a probability found from the cumulative distribution function of the hypergeometric density as

$$\mu = \sum_{i=0}^{c-1} \binom{n}{i} \binom{N-n}{d-i} \binom{N}{d}^{-1}. \quad (12)$$

If no good pairing is found up to time t , the number of completed tests will be

$$\eta = \lfloor t/(\tau + d) \rfloor. \quad (13)$$

If we assume that the algorithm blindly throws away all hypotheses that have not passed a complete test by the time t , P is simply the probability that some hypothesis passes the test before the time is up. Then

$$P = 1 - [1 + \lambda(\mu - 1)]^\eta, \quad (14)$$

from which we can compute the efficiency of the preemption scheme with $T_{c,d}$ test.

We will now turn to investigate preemption schemes of the type defined by Algorithm 1 in the *i.a.*-model. For a given time budget t we will determine a preemption function f that gives the maximum efficiency over all preemption functions. In order to accomplish this, we will observe two things. First, a fixed scoring sequence X_i is equivalent to any other scoring sequence that includes the same terms, regardless of order, provided that the scoring will not be stopped before the term indexed by i . Second, with Algorithm 1 in the *i.a.*-model, there is no reason to reorder the hypotheses before a good pairing has been found. More precisely, the probability of finding a good pairing is not changed by the reordering. Once a good pairing has been found, the scheme will catch the inlier hypothesis associated with that pairing and keep it or some other inlier hypothesis to the end. But since P is simply the probability that a good pairing will be found, we can think of the scheme as if it stopped directly with the first inlier hypothesis found. This means that we can ignore the reordering in Step 4 as if it was a null operation and the ordering was fixed from the outset. Taken together, these two facts mean that in the *i.a.*-model, the efficiency for a fixed time t of a scheme of type Algorithm 1 is equivalent to the efficiency of some preemption scheme with the following constraint on the scoring sequence X_i (and vice versa):

$$[(o, h) \in X_i] \Rightarrow [(j, k) \in X_i \mid j = 1, \dots, o, k = 1, \dots, h]. \quad (15)$$

Thus, we can equivalently maximize the efficiency of the latter type of scheme, which we call *block-dominant*. The first good pairing found by the block-dominant scheme must be the pairing of the first inlier observation with the first inlier hypothesis. Maximizing efficiency is thus equivalent to maximizing the probability that this pairing is present in the evaluated terms. Let the probability distribution of the first inlier observation be $p_O(o)$. Let the probability distribution of the first inlier hypothesis be $p_H(h)$. According to Assumption 4, these distributions are independent. Moreover,

$$p_O(o) = \begin{cases} \frac{n}{N-o+1} \binom{N-n}{o-1} \binom{N}{o-1}^{-1} & o \leq N-n+1 \\ 0 & o > N-n+1 \end{cases} \quad (16)$$

$$p_H(h) = (1 - \lambda)^{h-1} \lambda. \quad (17)$$

and the efficiency of the preemption scheme is

$$E(f) = n \sum_{o=1}^N \sum_{h=1}^{f(o)} p_{O,H}(o, h), \quad (18)$$

where $p_{O,H}(o, h) = p_O(o) p_H(h)$. We wish to find the most efficient preemption schemes for different times t . If we first assume that we know that $f(1) = M$ for some M , this is tractable. We start with the preemption scheme for which $f(1) = M$ and $f(o) = 0$ for $o > 1$. Then we inductively keep adding the unused term for which $p_{O,H}$ is the largest, with the restriction that $h \leq f(1)$. The functions p_O and p_H are decreasing, which means that we must end up with a valid block-dominant scheme. Moreover, all the schemes that we pass through must necessarily maximize the efficiency for the time that they spend over all preemption schemes for which $f(1) = M$. The optimal block-dominant schemes in the *i.a.*-model can therefore be computed for all t up to some value t_{\max} of choice by computing the optimal schemes under the restriction $f(1) = M$ for all $M \leq (t_{\max}/\tau)$. The execution time follows Eq. (8) and schemes for the same t are easily compared using Eq. (18).

Finally, we will just briefly indicate that in the *i.a.*-model, the most efficient scoring sequence starts out down the diagonal, i.e. $x_i = (i, i)$. Let us assume for a moment that the hypothesis generation cost is zero, i.e. $\tau = 0$. Let us also assume another probability distribution for the observations, where an observation is an inlier with a probability α independent of the other observations. Then, the maximum efficiency is achieved by the scoring sequence defined by $x_i = ((i - 1) \% N) + 1, i$, where $\%$ denotes the modulo operation. To see this, first observe that there can never be a

Table 1 The most important difference is that the relative hypothesis generation time τ , i.e. the time spent generating a hypothesis relative to the time spent evaluating a likelihood term, is much higher in the five-point case

	3 point	5 point
No. of observations, N	500	500
No. of inliers, n	250	250
Sample size, s	4	6
Relative hypothesis generation time, τ	12	250
Time budget, t	102,500	225,000
No. of samples, M	500	500
Block size, B	100	100
Total hypothesis generation time (ms)	0.1	60
Total scoring time (ms)	25	60
Preemption overhead (μ s)	100	100

The lower part shows the actual preemption parameters used in the real-time system. The last three rows give the actual hypothesis generation, scoring, and preemption overhead times of the real system on a modest 550 MHz machine. Note that the scoring time totally dominates the three-point case. The hypothesis generation takes half of the time in the five-point case, but without preemptive scoring, the scoring time can be five times as high. Thus, faster scoring directly improves the run times in both cases. Observe also that the preemption overhead is absolutely negligible.

Table 2 Approximate average timings per 720×240 frame of video for the system components on a modest 550 MHz machine

Feature detection	Matching with disparity range			
	3%	5%	10%	SaM
30 ms	34 ms	45 ms	160 ms	50 ms

MMX code was used for the crucial parts of the feature detection and feature matching. Disparity range for the matching is given in percent of the image dimensions. In the structure and motion component (SaM), one-view and three-view estimations are combined to incrementally build the reconstruction with low latency. The whole system including all overhead currently operates at 26 frames per second on average on a 2.4 GHz machine when using a 3% disparity range. The latency is also low, since there is no self-calibration and only very local iterative refinements.

reason for scoring a hypothesis twice, since we can generate a new hypothesis for free which has never participated in a bad pairing and therefore is an inlier with as high or higher probability. Let now $\gamma(o)$ denote the number of times observation o is paired with a new hypothesis in the scoring sequence. Then the probability of not finding a good pairing is

$$\prod_{o=1}^N (1 - \alpha + \alpha(1 - \lambda)^{\gamma(o)}). \quad (19)$$

Under the constraint $\sum_{o=1}^N \gamma(o) = t$, this has to be minimized by a function $\gamma(o)$ that is as constant as possible. To see this, observe that adding 1 to $\gamma(o)$ subtracts $\alpha\lambda(1 - \lambda)^{\gamma(o)}$ from the factor of index o , which in turn reduces the whole probability by the fraction

$$\frac{\alpha\lambda}{\alpha + (1 - \alpha)(1 - \lambda)^{-\gamma(o)}} \quad (20)$$

of its value. Since this fraction decreases with $\gamma(o)$, the statement follows.

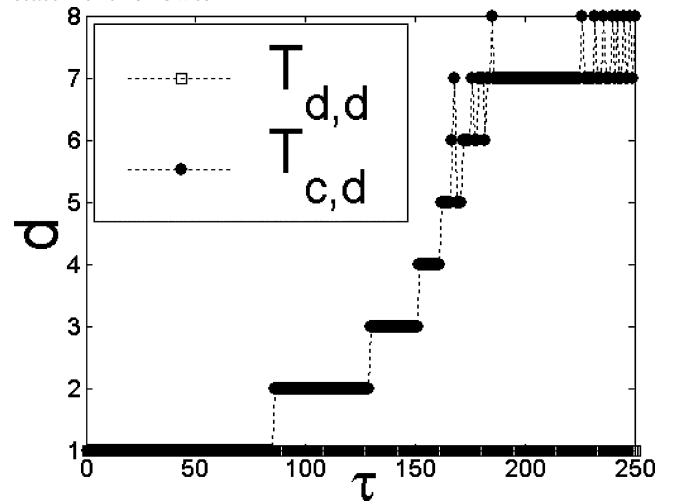


Fig. 3 The choice of d that leads to the highest efficiency for the $T_{c,d}$ test in the *i.a.*-model as the relative hypothesis generation time τ increases up to the value encountered in the five-point situation. We used $s = 4$, c turns out always to be one. When $\tau = 250$, the 1-out-of-8 test is the most efficient. The best choice of d for the $T_{d,d}$ test is always one

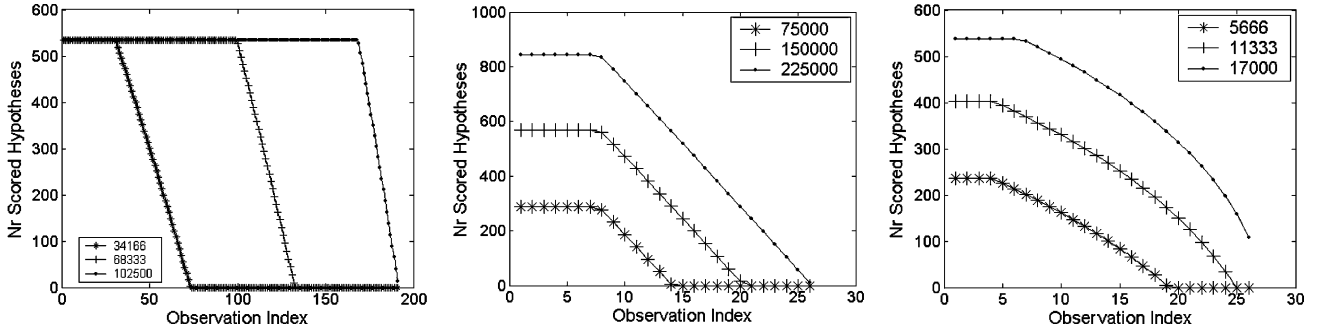


Fig. 4 The most efficient preemption functions f in the *i.a.*-model for three different parameter settings and three different time budgets. *Left*: The three-point case with budgets up to the one used in the real system. *Middle*: The five-point case with budgets up to the one used in the real system. *Right*: The three-point case with smaller budgets and a smaller number of observations ($N = 50, n = 25$). The decrease on the right in the number of scored hypotheses is not likely to be efficient in practice. Rather, a narrow tail of hypotheses that are scored against all observations should be favorable, since that gives more discrimination between a small set of good hypotheses. Thus, it is underlined that the *i.a.*-model is useful only for modeling the initial stage of scoring, where the goal is to weed out contaminated hypotheses. However, this stage is the main responsibility of the preemptive scoring, since iterative refinement will then take over

7 Results

We set the parameters of the *i.a.*-model to mimic the real system when relative pose is estimated with the five-point method as hypothesis generator and when absolute pose is estimated with the three-point method. The default parameters are shown in Table 1. The most important difference is that the relative hypothesis generation time τ is much higher in the five-point case. This is a less commonly encountered RANSAC situation, since hypothesis generation is seldom as complicated as the five-point method. Thus, improved scoring times achieved by preemptive scoring directly improves the run times. Observe that this is true even for the five-point situation. Average overall system timings are shown in Table 2.

When hypothesis generation becomes costly, the $T_{d,d}$ is not patient enough, since it becomes important not to throw away good motion hypotheses. Figure 3 shows the choice of d that leads to the highest efficiency for the $T_{c,d}$ test in the *i.a.*-model as the relative hypothesis generation time τ increases up to the value encountered in the five-point situation. c turns out always to be one. When $\tau = 250$, the 1-out-of-8 test is the most efficient.

Figure 4 shows the most efficient preemption functions f in the *i.a.*-model for three different parameter settings and three different time budgets. Note that all generated hypotheses are scored a number of times before the preemption starts abandoning hypotheses.

We have tested preemption using Algorithm 1 and the preemption function 9 on synthetic data. An example result is shown in Fig. 5. See the figure caption for more details.

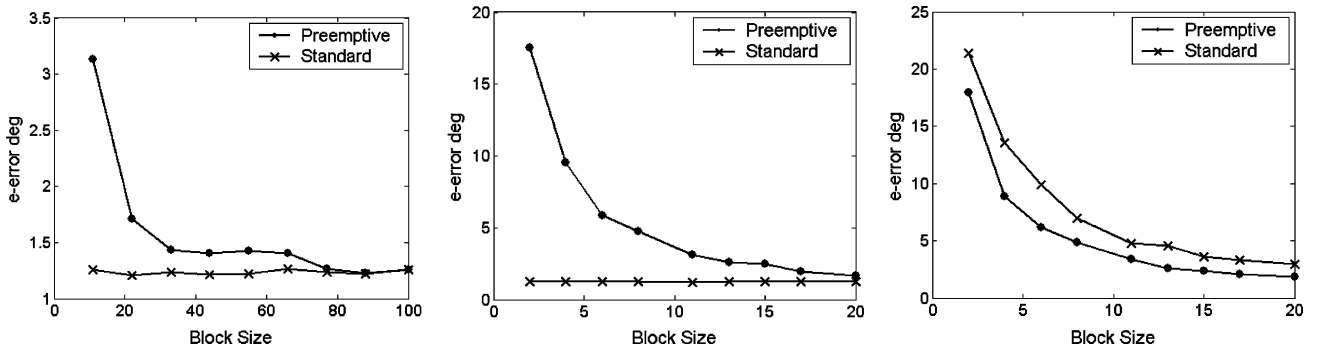


Fig. 5 The effect of preemption on the translational error in the relative orientation problem investigated with synthetic data. For this particular example, the baseline was 0.1 of the distance to the scene, which was 0.5 deep. Noise standard deviation was 1 pixel of a CIF image and field of view was 45° . One fifth of the observations was replaced by gross outliers moving coherently in an incorrect direction by a magnitude of 60 pixels. The translational error is given in degrees as the average over 1,000 trials. ‘Preemptive’ stands for Algorithm 1 with the preemption function 9. The number of samples were $M = 500$ and the block size B was varied. ‘Standard’ is the standard RANSAC scheme. *Left*: Here the block size was decreased in the preemptive algorithm until the error starts increasing. The standard scheme always performs full scoring. Note that we can introduce significant preemption without affecting the translational error very much. Our default block size of 100 used in the real-time system gives as good estimates as the standard scheme, only faster. *Middle*: The same situation with even heavier preemption (lower block size). *Right*: As mentioned previously, the only fair way to compare preemption schemes is to compare them for the same time budget. Here we show the standard scheme restricted to the same time budget as the preemptive version. The preemptive version is always better

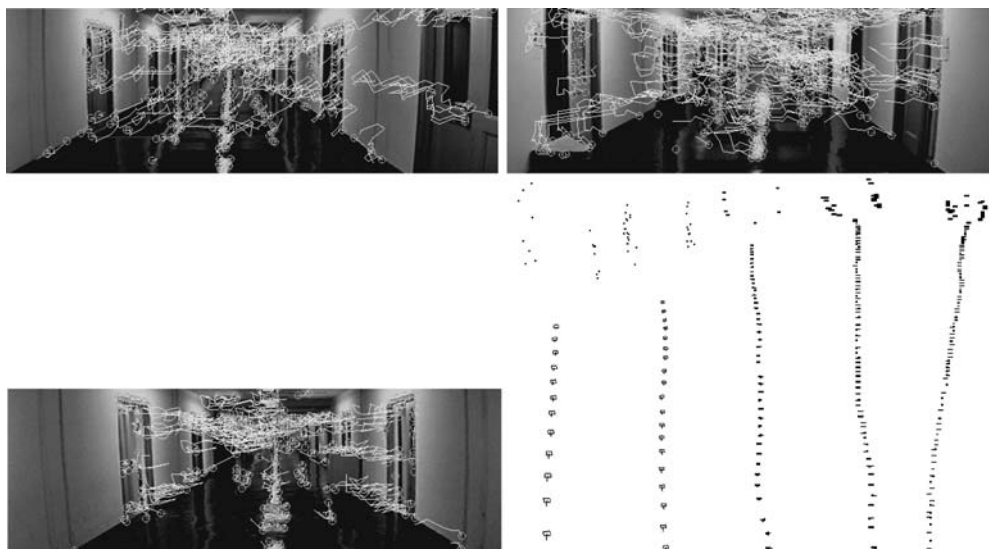


Fig. 6 Estimation of the motion of a hand-held camera, induced by walking down a 160 m long corridor. The whole motion is successfully integrated into the same coordinate frame. Note how the straight trajectory builds up, with different stages shown from *left to right*. The estimation was done in real-time from a camcorder tape playing back through the PC capture card. Thus, the system has to deal with capture, dropping frames if not able to keep up etc. The estimation was made with a graphical feature-track window and a 3D scene window displaying. In this mode, the delay from the tape to the reconstruction displaying on the screen is less than a second. No prior knowledge of the motion was used and the system is identical to the one used in Fig. 7

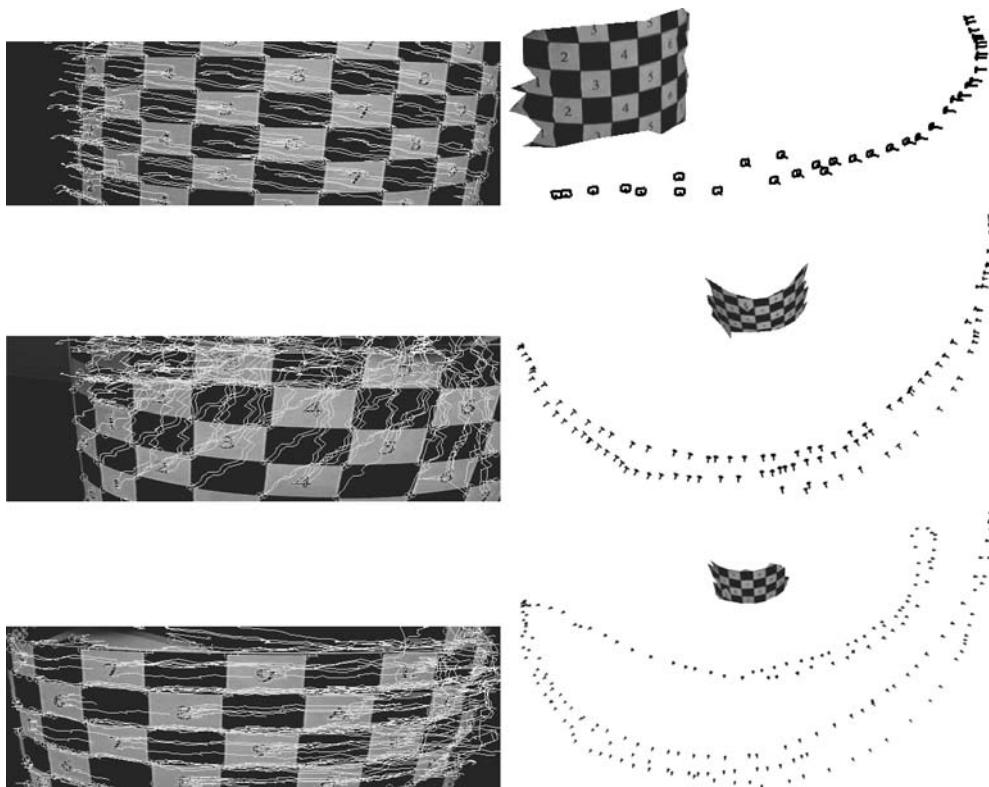


Fig. 7 Estimation of a freehand motion created with a calibration cylinder in one hand and a camcorder in the other. The estimation was done live. Thus, the system has to deal with capture, dropping frames if not able to keep up etc. The estimation was made with both a feature-track window and a 3D scene window displaying. Note how the trajectory builds up. No knowledge of the structure or motion was used and the system is identical to the one used in Fig. 6

The preemptive scoring enables us to perform structure and motion estimation in real-time. Some reconstructions made while loading the video source from disk are shown in Fig. 1. Results from live estimation, dealing with all issues like dropping frames when not keeping up etc. is shown in Figs. 6 and 7. No knowledge of the motion was used in either case and the system parameters were identical. This shows that the system can handle forward as well as sideways motion.

8 Summary and conclusions

A scheme for preemptive scoring in RANSAC was proposed. The problem of preemption was stated in a general fashion and was analyzed theoretically under a simple inlier-outlier model. The proposed preemption scheme was tested with synthetic data. Successful live and in real-time structure and motion estimation with the preemptive scheme was demonstrated on real data.

References

- Triggs, B., McLauchlan, P., Hartley, R., Fitzgibbon, A.: Bundle adjustment—a modern synthesis. Springer Lecture Notes on Computer Science, vol. 1883, pp. 298–375. Springer Verlag, Berlin Heidelberg New York (2000)
- Fischler, M., Bolles, R.: Random sample consensus: a paradigm for model fitting with application to image analysis and automated cartography. *Commun. Assoc. Comp. Mach.* **24**, 381–395 (1981)
- Torr, P., Zisserman, A.: MLESAC: a new robust estimator with application to estimating image geometry. *Comput. Vision Image Understand.* **78**, 138–156 (2000)
- Torr, P., Murray, D.: The development and comparison of robust methods for estimating the fundamental matrix. *Int. J. Comput. Vision* **24**(3), 271–300 (1997)
- Torr, P., Murray, D.: Outlier detection and motion segmentation. In: SPIE Sensor Fusion Conference VI, pp. 432–443 (September 1993)
- Zhang, Z.: Determining the epipolar geometry and its uncertainty: a review. *Int. J. Comput. Vision* **27**(2), 161–195 (1998)
- Beardsley, P., Zisserman, A., Murray, D.: Sequential updating of projective and affine structure from motion. *Int. J. Comput. Vision* **23**(3), 235–259 (1997)
- Nistér, D.: Reconstruction from uncalibrated sequences with a hierarchy of trifocal tensors. In: Proceedings of the European Conference on Computer Vision, vol. 1, pp. 649–663 (2000)
- Nistér, D.: In: Automatic dense reconstruction from uncalibrated video sequences. PhD Thesis, Royal Institute of Technology KTH, ISBN 91-7283-053-0 (March 2001)
- Pollefeys, M., Koch, R., Van Gool, L.: Self-calibration and metric reconstruction in spite of varying and unknown internal camera parameters. *Int. J. Comput. Vision* **32**(1), 7–25 (1999)
- Pollefeys, M., Verbiest, F., Van Gool, L.: Surviving dominant planes in uncalibrated structure and motion recovery. In: Proceedings of the European Conference on Computer Vision, vol. 2, pp. 837–851 (2002)
- Torr, P., Fitzgibbon, A., Zisserman, A.: The problem of degeneracy in structure and motion recovery from uncalibrated image sequences. *Int. J. Comput. Vision* **32**(1), 27–44 (1999)
- 2d3 Ltd. Boujou, <http://www.2d3.com>
- Oliensis, J.: A critique of structure from motion algorithms. *Comput. Vision Image Understand.* **80**, 172–214 (2000)
- Nistér, D.: An efficient solution to the five-point relative pose problem. In: IEEE Conference on Computer Vision and Pattern Recognition, vol. 2, pp. 195–202 (2003)
- Hartley, R., Silpa-Anan, C.: Reconstruction from two views using approximate calibration. In: Proceedings of the 5th Asian Conference on Computer Vision, Melbourne, Australia (January 2002)
- Björkman, M., Eklundh, J.: Real-time epipolar geometry estimation and disparity. In: Proceedings of the International Conference on Computer Vision, pp. 234–241 (1999)
- Chiuso, A., Favaro, P., Jin, H., Soatto, S.: 3-D motion and structure causally integrated over time: implementation. In: Proceedings of the European Conference on Computer Vision. Lecture Notes in Computer Science, vol. 1842, pp. 735–750. Springer Verlag, Berlin Heidelberg New York (2000)
- Jin, H., Favaro, P., Soatto, S.: Real-time 3-D motion and structure from point features: a front-end system for vision-based control and interaction. In: Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition, pp. 778–779 (2000)
- Jin, H., Favaro, P., Soatto, S.: Real-time feature tracking and outlier rejection with changes in illumination. In: International Conference on Computer Vision, pp. 684–689 (2001)
- Chum, O., Matas, J.: Randomized RANSAC with $T_{d,d}$ test. In: Proceedings of the British Machine Vision Conference, pp. 448–457 (2002)
- Tordoff, B., Murray, D.: Guided sampling and consensus for motion estimation. In: Proceedings of the European Conference on Computer Vision. Springer Lecture Notes on Computer Science, vol. 2350, pp. 82–96 (2002)
- Haralick, R., Lee, C., Ottenberg, K., Nölle, M.: Review and analysis of solutions of the three point perspective pose estimation problem. *Int. J. Comput. Vision* **13**(3), 331–356 (1994)
- Press, W., Teukolsky, S., Vetterling, W., Flannery, B.: Numerical recipes in C. Cambridge University Press, Cambridge (1988)
- Lacey, A., Pinitkarn, N., Thacker, N.: An evaluation of the performance of RANSAC algorithms for stereo camera calibration. In: Proceedings of British Machine Vision Conference, pp. 646–655 (2000)



David Nistér received PhD degree in computer vision, numerical analysis and computing science from the Royal Institute of Technology (KTH), Stockholm, Sweden, with the thesis 'Automatic Dense Reconstruction from Uncalibrated Video Sequences'. He is currently an assistant professor at the Computer Science Department and the Center for Visualization and Virtual Environments, University of Kentucky, Lexington. Before joining UK, he was a researcher in the Vision Technologies Laboratory, Sarnoff Corporation, Princeton, and Visual Technology, Ericsson Research, Stockholm, Sweden. His research interests include computer vision, computer graphics, structure from motion, multiple view geometry, Bayesian formulations, tracking, recognition, image and video compression. He is a member of the IEEE and American Mensa.