Technical Report

Motion Interpretation using Adaptive Search of Transformation Space

Adrian Ulges

ulges@iupr.net

IUPR Research Group
Faculty of Computer Science
Technical University Kaiserslautern

June 2007

Motion Interpretation using Adaptive Search of Transformation Space

Adrian Ulges
ulges@iupr.net
IUPR Research Group
DFKI

June 5, 2007

Abstract

This report addresses the extraction of a parametric global motion from a motion field, a task with several applications in video processing. We present two probabilistic formulations of the problem and carry out optimization using the RAST algorithm, a geometric matching method novel to motion estimation in video. RAST uses an exhaustive and adaptive search of transformation space and thus gives – in contrast to local sampling optimization techniques used in the past – a globally optimal solution. Among other applications, our framework can thus be used to generate ground truth for benchmarking motion estimation.

Our main contributions are: first, the novel combination of a state-of-the-art quality criterion for dominant motion estimation with a search procedure that guarantees global optimality. Second, experimental results that illustrate the superior performance of our approach on synthetic flow fields as well as real-world video streams. Third, a significant speedup of the search achieved by extending a basic model with an additional smoothness prior.

Contents

1 Introduction

We address the estimation of a dominant parametric motion from a sequence of video frames. Such dominant motion is usually equated with background motion, and its precise and robust estimation is required for several applications in the context of video analysis, like motion-based segmentation or motion compensation (which again serves as a building block in modern video encoders, or in video mosaicing).

Approaches to solve this problem can be divided into two categories: first, direct methods, which propose joint formulations for motion estimation and segmentation and usually include difficult, cost-worthy optimization procedures. Second, indirect methods that first estimate a motion field and then segment it. Though such indirect approaches are prone to inaccuracies in the motion estimation step and do not reach the robustness of direct methods, they offer simple, fast, and stable alternatives that are used in many practical video processing systems.

Our approach belongs to this category of indirect methods, i.e. we estimate a global parametric motion from a given field of local motion probes – a problem that is difficult due to measurement noise, inaccuracies of the previous motion estimation step, and deviant foreground motion. In terms of dominant motion estimation, such foreground motions are "outliers" that have to be recognized and discarded during the fitting process.

We view the problem from a parameter estimation perspective and propose two probabilistic formulations: one formulation that assumes independence of the single flow samples, and one imposing spatial coherence of motion using a smoothness prior (similar formulations can be found in the literature [15, 4, 20]).

The resulting optimization problems are solved using the RAST (*Recognition by Adaptive Subdivision of Transformation Space*) algorithm [2]. While other methods are based on a sparse sampling of search space and do not guarantee optimal solutions, RAST performs an adaptive but exhaustive search and finds the global optimum.

Our main contributions are: first, a novel dominant motion estimation approach that gives an optimal solution under a clearly defined statistical criterion. For example, our approach might be used to benchmark local search techniques on real-world video data, where motion fields are noisy and ground truth is not known. Second, experimental results on synthetic data and real-world videos that proof the superior performance of our framework compared to other local search procedures. Third, a novel extension to the RAST algorithm including a smoothness prior that leads to a more efficient search.

The remainder of this report is organized as follows: we first introduce related work in Section 2 before we present our approach in Section 3. Experimental results are outlined in Section 4, and finally a discussion is given in Section 5.

2 Related Work

The problem of dominant motion estimation is strongly related to motion segmentation: while the former estimates the motion with the largest support region (and the support region associated with it), the latter addresses a complete segmentation of the frame into two or more regions. Thereby, the solution to one problem inherently leads to a solution for the other: on the one hand, a complete motion segmentation trivially comes with a dominant motion (the one associated with the largest region). On the other hand, motion segmentation can be achieved by iteratively estimating the dominant motion and ignoring its support region in the following iterations. This process is repeated until the whole frame is explained [1] (in fact, the optimization procedure we propose follows a similar strategy within a global optimization process). It is due to this equivalence that we present related work to solve either of the problems under the term *motion interpretation*.

Such motion interpretation has often been called a "chicken-egg" problem: motion estimation is inaccurate without knowledge of motion boundaries due to the aperture problem [1], while on the other hand motion segmentation demands local motion estimates. Methods to solve this problem can be divided into direct and indirect (or "feature-based" [9]) methods. Approaches from the first category propose joint formulations for both estimating motion and grouping it into coherent regions. This usually leads to energy functions containing a "goodness-of-fit" term and a "smoothness" term. Some direct methods assume a parametric motion over image regions [1, 10, 19, 4, 15]. Others are non-parametric and based on piecewise smoothness of the motion field, which leads to formulations related to Markov Random Fields [14, 20].

Several procedures have been proposed to optimize the resulting energy terms: EM [20], graduated non-convexity [1], segmentation and grouping [19], or more recent formulations that alternately estimate motion and readjust the segmentation, using graph cuts [15] or level set methods [4]. All these approaches have in common that optimizing the associated energy is cost-worthy, prone to local minima, and sensitive to the chosen parameters.

In contrast to this, indirect methods separate motion estimation and segmentation. While direct approaches are often based on an iterative optimization, indirect ones are two-step procedures: first, an optical flow field is estimated using correlation-based techniques [18], feature tracking [17], or optical flow. The result forms the input to a segmentation step, which must cope with local outliers and inaccuracies due to noise in the measurement process, error-prone motion estimation, and foreground objects in motion. For this, greedy local search procedures have been used in the past, like robust least squares, RANSAC [5], least median of squares, or least trimmed squares [12].

Since local errors in the motion estimation step cannot be undone, indirect methods theoretically do not reach the robustness of direct ones. Nevertheless, they offer simple and fast alternatives that are popular in practice, and are applied to several video processing tasks, like in state-of-the-art video codecs or video mosaicing [16].

Our approach belongs to this category of indirect approaches. More precisely, we assume a motion field is given and focus on the motion interpretation step. We propose an optimization procedure based on a full, adaptive search of transformation space. While methods used in the past based their optimization on a sparse sampling that tends to get caught in local minima, our framework gives the global optimum.

3 Statistical Framework

We assume a given motion field $D = \{(x_1, v_1), .., (x_n, v_n)\}$ of 2D positions x_i associated with 2D motion vectors v_i, i.e. features at positions x_i in the first frame move to $x_i + v_i$ in the second one. This data can be a dense optical flow field, or sparse probes obtained from block matching or tracked point features. The task is now to extract a parametric motion $v_\theta : \mathcal{R}^2 \rightarrow \mathcal{R}^2$ that fits D "well", i.e. $\frac{\partial x_i}{\partial t} = v_i \approx v_\theta(x_i)$.

3.1 Motion Parametrization

In real-world videos, the image motion is a projection of the scene motion and depends heavily on the 3D structure of the scene. Consequently, the actual image motion cannot be recovered directly without scene knowledge. However, *approximations* can be made to the shape of the scene and the camera projection model that lead to *parametrized* motion models. Such parametrized motion has proven a simple and often sufficiently accurate approximation to projected 3D scene motion that is widely used in practice. We adopt this parametric approach and start with listing some typical parametrizations from the literature [9, 16]:

- **constant motion** - the background moves in a constant shift: $v(x_i) = v'$
- **similarity transform** the model consists of a rotation by an angle α, a scaling s (e.g., due to zooming), and a translation $(d_x, d_y)^T$. This model corresponds to the camera watching a planar surface that is perpendicular to the optical axis.

$$v(x_i) = \begin{pmatrix} s \cdot cos\alpha & -s \cdot sin\alpha \\ s \cdot sin\alpha & s \cdot cos\alpha \end{pmatrix} \cdot x_i + (d_x, d_y)^T$$

- **affine** - the 6-parameter affine model corresponds to a planar surface under orthographic camera projection [16] It is probably the most widely applied motion model, offering a good tradeoff between model complexity and accuracy:

$$v(x_i) = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \cdot x_i + (d_x, d_y)^T$$

- **8-parameter** - the assumptions underlying the 8-parameter motion model are a perspective projection (like in most real-world cameras) and a planar scene. The resulting mapping between successive frames is an 8-parametric *homography*:

$$v(x_i) = \frac{1}{c_1 x + c_2 y + 1} \begin{pmatrix} a_1 & a_2 & a_3 \\ b_1 & b_2 & b_3 \end{pmatrix} \cdot (x_i, y_i, 1)^T$$

- **parabolic** - the restrictions of the 8-parameter model can be loosened further by assuming the scene structure by a parabolic surface, obtaining

the nonlinear parametrization:

$$v(x_i) = \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ b_1 & b_2 & b_3 & b_4 & b_5 \end{pmatrix} \cdot (x, y, x^2, y^2, xy)^T + (d_x, d_y)^T$$

From these parametrizations, we choose the similarity transform as a good balance between model simplicity and accuracy.

3.2 Bayesian Formulation of the Problem

We view motion interpretation as a parameter estimation problem, i.e. the parameters θ of a dominant motion are to be found that fit D "well". The well-known MAP formulation of this problem is to choose the global motion $\hat{\theta} = (\hat{s}, \hat{\alpha}, \hat{d_x}, \hat{d_y})$ that maximizes the posterior:

$$\hat{\theta} = \arg\max_{\theta} \quad P(\theta|D) \propto P(D|\theta) \cdot P(\theta) \tag{1}$$

In the following, we derive two quality criteria from this generic formula. The first one assumes a uniform prior over all possible global motions leading to a maximum likelihood (ML) optimality criterion in which local motion samples are assumed independent. In contrast to this, the second formulation uses a prior to impose the constraint of *spatial coherence* with which motion occurs in real-world image sequences. Both criteria lead to different quality functions whose derivation will be described in the following.

3.2.1 Criterion 1: Local Independence

For our first formulation, we assume a uniform prior $P(\theta)$ and independent motion probes drawn from a distribution $p(v_i|\theta)$. If we also neglected competitive foreground motion and used isotropic Gaussian noise to model inaccuracies of motion estimation and of the capturing process, $p(v_i|\theta)$ would be a Gaussian with mean $v_\theta(x_i)$ and diagonal covariance $\sigma^2 I$:

$$p'(v_i|\theta) = \mathcal{N}(v_i; v_\theta(x_i), \sigma^2 I) \tag{2}$$

In practical flow fields, however, *outliers* occur – again, due to inaccuracies of the motion estimation process, but also due to foreground objects moving into a different direction. Since we do not have prior knowledge of the motion of such objects, we assume a uniform distribution $p(v_i|\theta) = c$ of foreground motion within a reasonable range. This gives a more realistic scenario, in which an observation is regarded as an outlier if it deviates too far from the dominant motion v_θ:

$$p(v_i|\theta) \propto \max\left(\mathcal{N}(v_i; v_\theta(x_i), \sigma^2 I), c\right) \tag{3}$$

We plug this term into the overall likelihood and obtain

$$p(D|\theta) = \prod_i p(v_i|\theta) \tag{4}$$

maximizing which is equivalent to maximizing the log-likelihood, which we again simplify further:

$$\begin{aligned}
\arg\max_\theta L(D|\theta) &= \arg\max_\theta \sum_i \log p(v_i|\theta) \\
&= \arg\max_\theta \sum_i \max\left(\log \frac{1}{2\pi\sigma^2} - \frac{1}{2\sigma^2}(v_i - v_\theta(x_i))^2, \log c\right) \\
&= \arg\max_\theta \sum_i \max\left(c' - \frac{1}{2\sigma^2}(v_i - v_\theta(x_i))^2, 0\right) \\
&= \arg\max_\theta \sum_i \max\left(1 - \frac{(v_i - v_\theta(x_i))^2}{\epsilon^2}, 0\right),
\end{aligned}$$

Thus, instead of maximizing the likelihood we equivalently maximize the simpler quality function:

$$Q_1(\theta) = \sum_i \underbrace{\max\left(1 - \frac{(v_i - v_\theta(x_i))^2}{\epsilon^2}, 0\right)}_{q(v_i,\theta)}. \tag{5}$$

Q_1 consists of local contributions $q(v_i, \theta)$ from the single flow samples, which are in the following referred to as the *support* of a local flow probe v_i for a global motion θ. It is zero exactly if v_i deviates further than ϵ from the model motion $v_\theta(x_i)$ (or if v_i is regarded as an outlier, respectively). Thus, the evaluation of Q_1 provides a segmentation of the motion field into background and foreground.

The only free parameter, ϵ, depends on the constant c and the expected noise σ^2. It determines the allowed deviation of a background motion sample from the parametric motion v_θ. In practice, this parameter is set manually.

Another fact worth noticing is that, since the overall support of a motion Q_1 is directly related to the likelihood, it can be used as a quality measure for a global motion $\hat{\theta}$. This fact will be important in the experimental section, where we judge the performance of several approaches based on the value of Q_1 they achieve.

3.2.2 Criterion 2: Spatial Coherence

The optimality criterion Q_1 introduced in Equation (5) is derived from the likelihood and neglects the spatial coherence with which motion occurs in real-world videos. Like other researchers before, we use this fact by formulating an additional prior related to formulations in Markov Random Fields [1, 7, 20].

For this, we first introduce a segmentation as a *labeling* of the motion vectors $L : \{x_1, .., x_n\} \to \{0, 1\}$ such that $L(x_i) = L_i = 1$ iff v_i belongs to the background (which is the case exactly if $q(v_i, \theta) > 0$). Note that – given such a labeling – we can automatically compute a motion estimate $\theta(\hat{L})$ as the least squares solution over the motion probes in the background region $L^{-1}(1)$. This

is why – instead of searching for a motion θ – we instead search for an optimal labeling by maximizing the posterior:

$$\begin{aligned} P(L|D) &\propto P(D|L) \cdot P(L) \\ &= P(D|\theta(L)) \cdot P(L) \end{aligned} \tag{6}$$

The first term corresponds to the likelihood criterion from Equation (5). For the prior $P(L)$, we define a neighborhood structure over the motion field sites $\{x_i\}$ (in this report, we use 4-connectedness on a regular grid of sites x_i), which again induces *cliques* of neighbor sites (in this report, all pairs of sites (x_i, x_j) that are adjacent). Let $\mathcal{C}$ denote the set of all such cliques. Then we define $P(L)$ as:

$$P(L) \propto \prod_{(x_i,x_j)\in\mathcal{C}} p(L_i, L_j) \tag{7}$$

which we rewrite further by replacing

$$p(L_i, L_j) = \begin{cases} \hat{c}_1 & L_i = L_j = 1 \\ \hat{c}_2 & \text{else} \end{cases} \tag{8}$$

with $\hat{c}_1 > \hat{c}_2$ (note that for the prior $P(L)$ to take on its maximum, the whole screen belongs to the background). By setting $c_1 = -\log \hat{c}_1$, $c_2 = -\log \hat{c}_2$ (with $c_2 > c_1$), and

$$U(i,j) = \begin{cases} c_1 & L_i = L_j = 1 \\ c_2 & \text{else} \end{cases} \tag{9}$$

the prior turns out to be:

$$P(L) \propto \prod_{(x_i,x_j)\in\mathcal{C}} e^{-U(i,j)} \tag{10}$$

This leads to the overall posterior for a labeling L and its associated motion θ:

$$P(L|D) \propto \prod_i p(v_i|\theta) \cdot \prod_{(x_i,x_j)\in\mathcal{C}} e^{-U(i,j)} \tag{11}$$

which we can again rewrite to obtain another Bayesian quality function:

$$\begin{aligned}
P(L|D) &\propto \prod_i p(v_i|\theta) \cdot \prod_{(x_i,x_j)\in\mathcal{C}} e^{-U(i,j)} \\
&\propto \prod_i \max(\frac{1}{Z} e^{-\frac{(v_i - v_\theta(x_i))^2}{2\sigma^2}}, c) \cdot \prod_{(x_i,x_j)\in\mathcal{C}} e^{-U(i,j)} \\
&\hat{=} \sum_i \max(-\frac{(v_i - v_\theta(x_i))^2}{2\sigma^2}, c') + \sum_{(x_i,x_j)\in\mathcal{C}} -U(i,j) \\
&\hat{=} \sum_i \max(c' - \frac{(v_i - v_\theta(x_i))^2}{2\sigma^2}, 0) + \sum_{(x_i,x_j)\in\mathcal{C}} (c_2 - c_1) \cdot L_i L_j \\
&\hat{=} \sum_i \max(1 - \frac{(v_i - v_\theta(x_i))^2}{\epsilon^2}, 0) + \alpha \sum_{(x_i,x_j)\in\mathcal{C}} (c_2 - c_1) \cdot L_i L_j \\
&\hat{=} \sum_i \max(1 - \frac{(v_i - v_\theta(x_i))^2}{\epsilon^2}, 0) + \alpha\beta \sum_{(x_i,x_j)\in\mathcal{C}} L_i L_j \\
&= Q_1(\theta) + \gamma \sum_{(x_i,x_j)\in\mathcal{C}} L_i L_j \qquad (12)
\end{aligned}$$

which gives us an MAP quality function

$$Q_2(\theta) = Q_1(\theta) + \gamma \sum_{(x_i,x_j)\in\mathcal{C}} L_i L_j \qquad (13)$$

Q_1 is the quality criterion from Equation (5). The free parameter $\gamma = \alpha\beta$ with $\alpha = 1./c'$ and $\beta = (c_2 - c_1)$. γ determines the weight of spatial coherence relative to the goodness-of-fit term Q_1 and is set manually in practice.

3.3 Optimization using RAST

Both quality functions Q_1 and Q_2 can be highly non-convex for motion fields in practice, such that techniques based on a sparse sampling of the space of possible motions $\{\theta\}$ may get caught in local minima. For example, robust least squares techniques start from a least squares solution θ_0 and iteratively refine the supporting local motions by discarding outliers. This gives a walk $\theta_0, .., \theta_t$ through transformation space that might contain the global optimum but is not guaranteed to.

We present an alternative based on a full search of parameter space. Though more time-consuming, it is made feasible using an *adaptive* search strategy. Our approach is called RAST (*Recognition by Adaptive Search of Transformation space*)[1]. It has been applied before in the domain of geometric matching and object recognition, where it is used to compute an optimal transformation to

[1] open source implementation at http://www.iupr.org/~chl/multirast.tar.gz

align point features. Our idea is to view each local motion probe as a correspondence between point features in two images (namely, x_i and $x_i + v_i$) such that RAST can be used to find an optimal transformation (or global motion, respectively).

In the following, the main features of the method are briefly outlined. More detailed descriptions of the algorithm can be found in [3, 2].

RAST is based on a branch-and-bound strategy: starting with the full parameter space $\{\theta\}$, a parameter subset θ' (also refered to as a *state*) is iteratively chosen and subdivided into subspaces θ_1, θ_2 by splitting along one parameter. We obtain subsequently finer subsets until finishing with a sufficiently small state whose center is returned as our estimate $\hat{\theta}$. The user can define the accuracy of the solution via this stopping criterion.

The search is guided into promising regions of parameter space by managing substates in a priority queue sorted by an upper bound $\mathcal{U}(\theta')$ for the quality in each state θ'.

$$\mathcal{U}(\theta') \geq \max\{Q(\theta)|\theta \in \theta'\}. \tag{14}$$

All substates are kept in a priority queue sorted by $\mathcal{U}$. In a branch-and-bound manner, RAST thus focuses on promising areas of transformation space.

The algorithm is subsumed here:

Algorithm 1 RAST

```
insert the full parameter space {θ} into the queue q
repeat
  extract the first element θ' from q
  split θ' into substates θ1, θ2
  compute U(θ1) and U(θ2) and insert θ1, θ2 into q
until θ' is small enough
```

We demand two properties of $\mathcal{U}$: first – as just stated – it must be a correct upper bound. Second, it must converge against the actual quality Q as the size of $|\theta'| \to 0$. Under these conditions, it can be shown that $\hat{\theta}$ returned by RAST is in fact the global optimum. Note that any function Q can be optimized using this generic scheme.

The key part of the search is the computation of $\mathcal{U}$, which is performed for each new subset to be inserted into the priority queue. For Q_1, the associated bound is $\mathcal{U}_1 = \sum_i u_i$, i.e. for each motion probe we find out (e.g., using interval arithmetic [3]) if it can contribute to *any* global motion in the subset. For Q_2, $\mathcal{U}_2 = \mathcal{U}_1 + \gamma \cdot \sum_{(i,j)\in\mathcal{C}} u_{ij}$ with $u_{ij} = 0$ if $u_i = u_j = 0$ and $u_{ij} = 1$ otherwise, i.e. after computing $\mathcal{U}_1$, an additional linear sweep through the motion probes is required to increment the bound for each pair of adjacent potential background sites.

As already mentioned in Section 2, the problem of dominant motion estimation is strongly related to motion segmentation. Given an approach for dominant motion estimation, a complete segmentation can be obtained by iteratively esti-

mating the dominant motion and discarding the motion probes associated with it.

With our framework, we can follow a similar approach and obtain a complete segmentation of the frame. Therefore, we do not have to start the RAST estimation multiple times – we simply carry out one global RAST optimization that does not stop when reaching a first optimum, but keeps searching until the whole frame has been explained.

4 Experiments

The most important capability of our approach is its optimality: the combination of our statistical framework and the RAST optimization guarantees an optimal solution given a complex statistical model. In a first set of experiments, we validate this fact on synthetic motion fields. These experiments provide a controlled framework for evaluation with a well-known ground truth segmentation and ground truth motion.

The second goal of this section is to point out the applicability of our method as a generator of ground truth for evaluating other motion estimation methods. Such an evaluation is a difficult task: visual inspection is not suitable for larger amounts of data, and ground truth motion and segmentation are usually not available. Also, the optimal performance is unknown under noisy and error-prone motion samples. In this situation, we suggest that our framework can serve as a source of ground truth.

Therefore, we present results on real-world video data. We validate that the RAST optimization procedure gives superior results to several local search procedures. For this comparison, we use the support of a global motion estimate in terms of the quality functions Q_1, Q_2 (which are directly related to the log-likelihood). It turns out that the performance of our framework does in fact provide an upper bound for the performance of other approaches.

Unfortunately, comparing motion support does not give a meaningful comparison of our two quality functions Q_1, Q_2 (trivially, $\forall\theta : Q_2(\theta) \geq Q_1(\theta)$). Therefore, we also compare motion segmentation results given by our framework for sequences with a known ground-truth segmentation.

4.1 General Setup

All input motion fields – synthetic or extracted from video – are defined at 16×16 macroblock positions (though our approach is not restricted to this setup). For video streams, motion is estimated using the MPEG-4 video codec XViD[2] [18]. Global motion is parametrized using a similarity transform. The following methods are tested:

1. *Our Framework*: We test our framework for both quality functions Q_1 and Q_2 ($\epsilon = 2.3$, $\gamma = 1$). The 4-dimensional similarity transform space searched by RAST should contain all reasonable motion between adjacent video frames. We choose: $\sigma \in [0.9, 1.1], \alpha \in [-0.1, 0.1], (d_x, d_y) \in [-40, 40]^2$. Search is stopped if the evaluated substate has dimensions smaller than $(0.0002)^2 \times (0.1)^2$. This means, the solution is determined with an accuracy of 0.1 pixels for the translation, or 0.0002 for the rotation and scale.

   Also, we add a least squares refinement over the background motion probes

[2] www.xvid.org

at the end given a sufficiently small substate θ':

$$\hat{\theta} := \arg\min_{\theta} \sum_{v_i : \exists \theta \in \theta' : q(v_i, \theta))) > 0} (v_i - v_\theta(x_i))^2 \tag{15}$$

2. *Least Squares*: A standard method to solve regression problems is given by least squares techniques. Here, the dominant motion θ is estimated by minimizing an error function E_D using methods from linear algebra [6]:

$$\hat{\theta} = \arg\min_{\theta} E_D(\theta) = \sum_i (v_i - v_\theta(x_i))^2 \tag{16}$$

This is equivalent to maximizing a quality function similar to Q_1, but with a pure Gaussian motion vector density instead of a truncated Gaussian one. Least squares is thus expected to perform poorly when competitive foreground motion occurs and serves as a baseline.

3. *Robust Least Squares*: robust least squares methods alternately compute least squares motion estimates and discard motion samples from D that deviate further from the solution than an outlier threshold σ. Our implementation generates a sequence of gradually non-convex solutions by decreasing σ. Its pseudocode is:

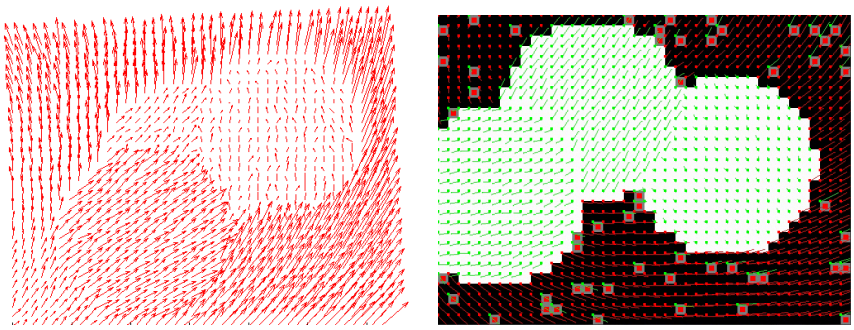
Algorithm 2 Robust Least Squares

set $k = 0$, $\sigma_0 = 100$, and D_0 to D
repeat
  set $\theta_k = \arg\min_\theta E_{D_k}(\theta)$
  set $D_{k+1} = \{x_i \in D_k | (v_i - v_{\theta_k}(x_i))^2 < \sigma_k\}$
  set $\sigma_{k+1} = 0.95 \cdot \sigma_k$
  set $k = k + 1$
until $(\sigma_k < \epsilon)$
return θ_k

4. *RANSAC*: Random Sample Consensus (RANSAC) [5] is a popular Monte Carlo procedure with excellent robustness to outliers and noise. The method is popular for parameter estimation in stereo vision [8] and tracking [13]. RANSAC is a stochastic algorithm: the solution is obtained by iteratively sampling a random subset $D^k \subset D$ consisting of k samples (since we estimate 4 parameters, two points are sufficient), estimating a least squares solution

$$\theta_k := \arg\min_{\theta} E_{D^k}(\theta) \tag{17}$$

on this subset (which corresponds to the assumption that D^k contains no outliers), and evaluating the quality of θ_k on the whole motion field D. This process is repeated K times, and the best estimate is returned. The probability of failure decreases with the number of iterations K, but never



(a) (b)

Figure 1: (a) A synthetic motion field with three blobs each moving in a different direction. (b) A segmentation result. Red vectors are assigned to the background, green ones to the foreground. Red blocks are segmentation errors that occur due to motion outliers.

reaches 0 – in contrast to our framework, optimality is not guaranteed. RANSAC is tested for both Q_1 and Q_2.

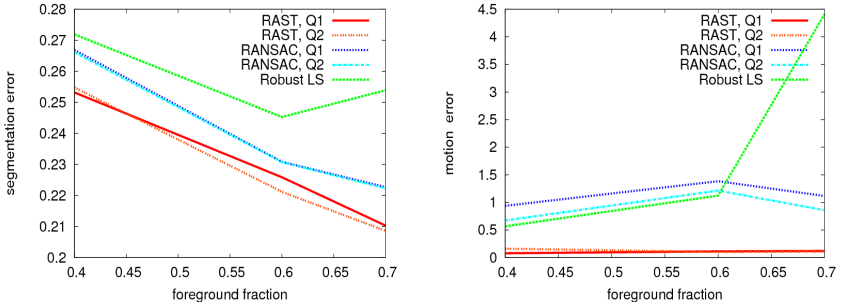
5. *XViD Dominant Motion Estimation*: the dominant motion estimation component that the XViD codec uses for compression purposes. The implementation is comparable to robust least squares, but with a more greedy outlier rejection strategy by thresholding with a residual.

4.2 Synthetic Flow Fields

In a first experiment, our purpose is to simulate the real-world phenomena of noise and spatial coherence of motion in a controlled setup. Therefore, we use synthetic flow fields of blob regions moving in front of a background moving as well.

Like the example illustrated in Figure 1(a) all motion fields are derived from a dominant motion and three foreground motions. The background motion is randomly drawn from $[-0.05, 0.05] \times [0.95, 1.05] \times [10, 10]^2$. Also, three blobs are initialized with a random motion from $\{0\} \times \{1\} \times [-16, 16]^2$. All blobs are of the same size such that they – when non-overlapping – occupy a certain fraction of the screen $f \in \{0.4, 0.6, 0.7, 0.8\}$. This fraction is varied throughout the experiment – the higher f, the more difficult the estimation of the correct dominant motion. Also, isotropic Gaussian noise with standard deviation $\sigma \in \{1.0, 1.3, 1.6, 2.0, 2.3\}$ is added to each motion vector. We generate five blob motion field sequences of 10 frames each for all combinations of noise levels σ and screen fractions f, obtaining a total of 1000 motion fields.

We run all test methods except the XViD codec (which we apply to real-world videos only) and least squares (which performed much worse than all other methods) on all motion fields. A typical result is given in Figure 1(b):



(a) (b)

Figure 2: Motion estimation results on synthetic blob data. (a) shows the average segmentation error (depending on the fraction of the screen occupied by competitive foreground motion), (b) shows the squared error of the estimated x translation relative to the ground truth.

vectors assigned to the background are red, vectors assigned to the foreground green. Also, blocks are colored depending on whether the segmentation result equals the ground truth information: for white (black) blocks, both assign the block to the foreground (background). For gray blocks (which are marked red additionally), they disagree, and a segmentation error occurs (in the illustration, such outliers are due to strong noise added to the local motion probes).

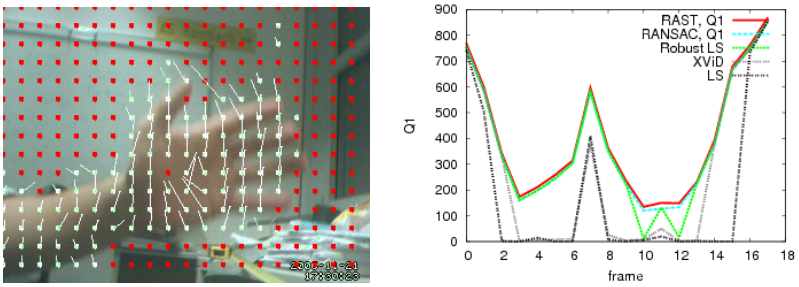
Numerical results are given in Figure 2. In Figure 2(a), the average segmentation error is plotted against the fraction f occupied by the foreground, reaching from 0.4 to 0.7. Note that some intrinsic segmentation error results from outliers due to noise. The rate of such outliers – and thus the segmentation error – constantly drops with f. As can be seen, our framework gives lower segmentation error rates than all other methods. The robust least squares method tends to break at high foreground fractions. Between RAST and RANSAC (100 iterations), a difference of about 1 % in segmentation error can be observed.

In Figure 2(b), we plot the average error of the estimated motion (more precisely, for the x-translation parameter) for the noise level $\sigma = 2.0$ against the foreground fraction f. Again, our framework shows the best performance. The average mean squared error remains below 0.2 pixels. RANSAC gives an average error in the order of 1 pixel. For the robust least squares method, however, the error increases strongly with f. This corresponds to the high segmentation error in Figure 2(a). Also, it can be observed that Q_1 and Q_2 give a similar performance regardless of the optimization procedure.

4.3 Test Sequence "Hand"

In a first test for real-world video data, a Firewire webcam[3] with a resolution of 320×240 pixels was used capturing a static scene. A hand was moved

[3]UniBrain Fire-I



(a) (b)

Figure 3: (a) A frame from a test sequence, with MPEG motion vectors plotted. Outliers are white, points contributing to the global motion are red. (b) The motion support for several test methods, plotted over the frames of the sequence.

at a distance of $35cm$ from the camera. See Figure 3(a) for a typical frame with results given by our approach: red vectors correspond to the global motion estimated, white ones to outlier motion. Obviously, the zero background motion was estimated correctly, though some outliers in the motion field occur due to error-prone motion estimation.

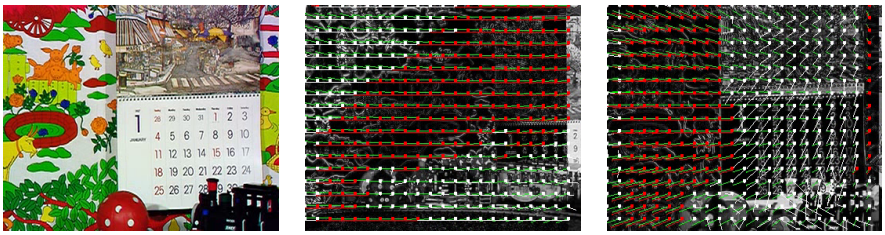
The support Q_1 for several test methods (distance $= 35cm$) is plotted against the frame number in Figure 3(b). Besides least squares, the XViD motion estimation shows the most breakdowns. Also, robust least squares fails at critical frames where the dominant motion is weak (as is indicated by a low support for all methods). RANSAC (100 iterations) performs comparable to RAST, which is to be expected as noise in this example is completely absent (due to specifics of the encoding process, background motion vectors tend to be exactly zero).

4.4 Test Sequence "Mobile"

In a scenario similar to the one in the last section, we apply our framework to MPEG-4 motion vectors derived from the "mobile and calendar" test sequence[4]. The sequence shows a textured background behind three foreground objects, each moving in a different direction approximately perpendicular to the optical axis. During the sequence, the camera zooms out and pans such that the dominance of the background motion increases gradually. We subsampled the sequence at 1 fps obtaining 11 frames with 22×18 macroblocks each.

One frame is shown in Figure 4 together with its motion estimates for XViD (Figure 4(b)) and RAST (Figure 4(c)). The motion visualization is layed over a difference image that is obtained by compensating with the estimated global motion and then subtracting the subsequent frame. For a perfect result, the difference should be zero (assuming constant pixel color) except for foreground regions. This is approximately the case for the RAST result. For XViD, on the

[4] http://www.m4if.org/resources.php



(a) (b) (c)

Figure 4: (a) A frame from the mobile sequence. Also: motion segmentation (red vectors belong to the background, white ones to the foreground) and difference between motion-compensated frames for XViD (b) and RAST (c). For XViD, a wrong estimate leads to a poor motion compensation on the upper left part of the frame.

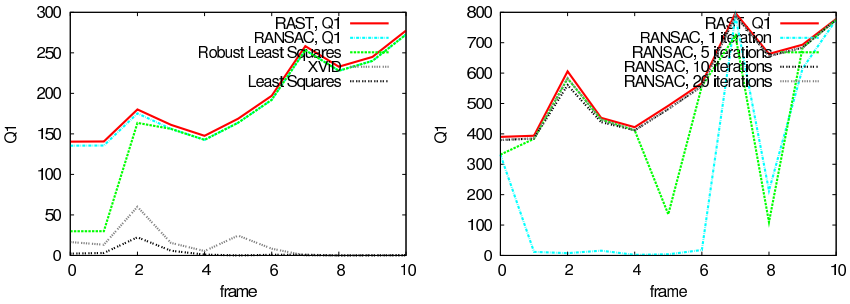
other hand, it can be seen that parts of the background (on the upper left) have been classified as foreground and have thus been poorly compensated for.

Figure 5(a) illustrates the motion support Q_1 for several test methods, plotted over the frames of the mobile test sequence. The support (and thus the dominance of the global motion) tends to be lower in the first frames of the test sequence, which corresponds to the pan operation towards the background wallpaper.

It can be seen that the RAST result serves as an upper bound for the performance of all other search procedures. Robust least squares gives comparable results except for the first two frames. RANSAC performs identical to RAST.

We also compared the average processing time of the two RAST methods for both criteria Q_1 (2.85 sec./frame) and Q_2 (1.07 sec./frame). Interestingly, the spatial prior – though demanding an extra sweep through all motion samples for the evaluation of a substate – leads to a significant speedup (62 %) that can be observed throughout all of our experiments. Obviously, spatial coherence helps to discard bad motion hypotheses early that are scattered over the frame, and guide search into promising regions of transformation space. This insight might be interesting in the geometric matching domain where RAST was developed.

As already mentioned before, the evaluation of dominant motion estimation is a difficult task for real-world video. The extracted motion fields may be error-prone and noisy such that it is unclear whether a performance measured in terms of motion support is a good performance given a motion field. In this context, our framework (which guarantees an optimal solution) can generate ground truth data to benchmark local search procedures. Here, we illustrate a simple sample experiment in which our framework is used to tune the number of iterative restarts K for the RANSAC algorithm. Theoretical considerations exist for this parameter given known outlier rates, where it is possible to estimate the number of iterations necessary to reduce the probability of error to a certain level. For real-world video data, however, noise and outliers make the situation less predictable. Instead, it is more practical to empirically estimate



(a)

Figure 5: (a) Motion support results for the frames of the subsampled mobile test sequence (11 frames (b) RANSAC results for the mobile sequence using different numbers of iterations. The quality of the estimate increase with the number of iterations, reaching its optimum at 20 iterations.

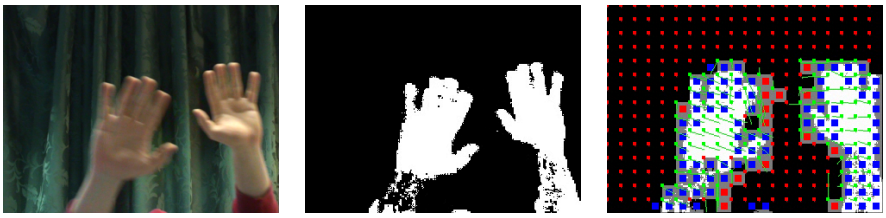
(a) (b) (c)

Figure 6: (a) A frame from the Snooker Sequence (b) its motion estimate (c) quality of several methods in terms of motion support for the snooker sequence (90 frames).

a "good" number of iterations for real-world video sequences. Therefore, we run our framework once before tuning starts, and tuning can be stopped as soon as RANSAC has reached the optimal performance. Our evaluation for RANSAC measures the average support Q_2 obtained over 10 random restarts and compared to the RAST support. Figure 4.4 shows the support plotted against the frame number for several values of K. The quality of the RANSAC estimates gradually increases with K, and has reached the RAST level at 20 iterations.

4.5 Test Sequence "Snooker"

Comparable results can be observed for a 640×480 test sequence called "snooker" captured from a TV sports broadcast. A snooker player is tracked by a camera with a strong translation. We test all methods on 90 frames (for RANSAC, 20 iterations were used). A sample frame is illustrated in Figure 6, with a RAST motion estimation result on the right visualized using a motion-compensated frame differencing as for the mobile sequence in Figure 4. It can be seen how motion compensation covers the background region, while differences for player



(a) (b) (c)

Figure 7: (a) Self-generated test data: two hands moving in front of the static background, with the camera performing zoom and pan operations. (b) Ground truth segmentation mask based on a skin color detector. (c) a segmentation result. Blue blocks are ignored, red blocks are misclassified.

and queue in the foreground occur.

The support Q_1 for the sequence is plotted in Figure 6(c). Again, XViD and least squares give relatively poor results. RANSAC and robust least squares perform comparable to our method, but fail occasionally. It can be seen how the support for our approach serves as an upper bound for other methods.

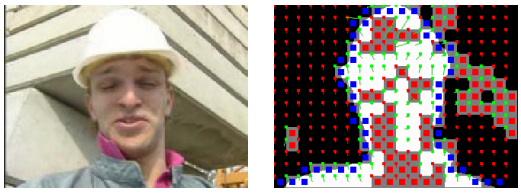
4.6 Test Sequence "Green Curtain"

In our last two experiments, we test whether our approach can be used for a motion-based segmentation of scenes. Therefore, we run tests on video sequences with a known ground truth segmentation. In our first segmentation experiment, we use self-generated video sequences as illustrated in Figure 7. Two hands move independently in front of a static green curtain. During the shots, the camera performed pan and zoom operations. Six sequences were generated of 125 frames each.

Ground truth segmentation masks were based on color using a histogram-based skin color model [11]. The resulting segmentation masks are not perfect, but sufficient for our purposes. We fuse them to block-level masks and ignore blocks that contain more than 5 % of both foreground and background pixels. All test methods were run on the 750 frames using both quality functions Q_1, Q_2 for RANSAC and RAST.

A segmentation result for a sample frame is presented in Figure 7(c). There are many "unsure" blocks (blue) partially overlapped by the hand, and some error blocks highlighted in red. Two main reasons for intrinsic errors could be made out: first, error-prone motion estimation (this can be observed in Figure 7(c), where error blocks in red correspond to outlier motion vectors). Second, errors occur for frames in which the object stands still for a moment.

The numerical results in terms of segmentation error rates are given in Table 1 and correspond to the performance observed in the previous test sequences, with our framework providing the optimal performance. A new insight is that the segmentation allows us to directly compare the performance of our quality



(a) (b)

Figure 8: (a) a frame from the foreman sequence, and (b) a typical segmentation result evaluated using MPEG-4 ground truth segmentation masks. Blue blocks are ignored, red blocks are misclassified.

criteria Q_1 and Q_2, where a slight increase in performance can be made out by including a spatial prior. This holds for both RAST and RANSAC.

4.7 Test Sequence "Foreman"

In the second segmentation experiment, we test the performance of our approach for motion segmentation in a real-world video. We use a subsampled version of the MPEG-4 test video sequence "foreman" (80 frames) that comes with a ground truth segmentation mask. The sequence shows strong, chaotic camera motion and a highly non-planar background.

Again, we tested several methods. For RAST and RANSAC (100 iterations) the spatial prior was included (Q_2). Segmentation results are compared to the ground truth on block basis (mixed blocks showing more than 5 % of both foreground and background pixels are ignored). The resulting error rates are given in Table 1. Our method gives the best results, followed by RANSAC and robust least squares. For both quality measures, no significant difference in performance can be observed.

A sample segmentation is illustrated in Figure 8. By visual inspection of these results, a high intrinsic error rate was found due to two reasons (besides inaccuracies in the motion estimation step): first, the object stands still in some frames and is missed by motion segmentation. Second, the 4D motion model implicitly assumes a planar background surface perpendicular to the optical axis. Since this assumption is heavily violated in the foreman sequence, the optimal motion fit cannot be determined in some frames.

5 Discussion

We have presented a framework for the indirect estimation of a global motion from a given motion field. Our method is based on two alternative probabilistic formulations of the problem: an ML criterion assuming independence of local motion samples, and an extension with a spatial coherence prior enforcing piecewise-smooth motion. The optimization of the resulting quality functions is done using RAST, an algorithm novel to dominant motion estimation in video.

The most important capability of our framework is that our method – in contrast to local search procedures used in the past – guarantees an optimal solution in terms of a clearly defined statistical criterion. We demonstrate this superior performance on synthetic motion data showing blobs moving in front of a noisy background motion, as well as on several real-world video sequences. Though greedy search procedures may be fast, attractive solutions for online processing, they do not guarantee global optimality. In this context, our framework can provide ground truth for benchmarking global motion estimation in video.

Another novelty we present is the combination of RAST optimization with a spatial prior formulation. In our experiments, we measured both a slightly better performance and a significant speed-up using this extension. Obviously, this approach helps to guide the adaptive RAST search into more promising regions of parameter space – an insight that might be interesting for traditional RAST applications in the area of geometric matching and object recognition.

Table 1: Average segmentation error rates for 700 frames of self-generated video using skin color ground truth (left) and for the "foreman" sequence (right). Blocks showing parts of both foreground and background were ignored.

method	segmentation error"green curtain" (%)	segmentation error "foreman" (%)
`RAST, Q1`	7.7	24.2
`RAST, Q2`	7.5	24.2
`RANSAC, Q1`	7.7	24.9
`RANSAC, Q2`	7.7	25.0
`Robust Least Squares`	7.8	25.5
`XViD`	11.7	33.1
`Least Squares`	40.6	41.4

Our experimental results on real-world scenes also point out how motion-based segmentation is possible using indirect motion estimation. However, the inherent limitations of the approach should be stated clearly: first, the motion estimates depends on the accuracy of local motion probes, which can be error-prone due to lack of texture, local optima in optimization and motion discontinuities. Second, the performance is poor if the motion does not fit the 4D

similarity transform model used, which happens for example with highly non-planar backgrounds near to the camera. Using higher-order motion models as outlined in Section 3.1 is generally possible with our framework, but is expected to come with higher cost, since a larger parameter space needs to be searched.

References

[1] M. J. Black and P. Anandan. The Robust Estimation of Multiple Motions: Parametric and Piecewise-Smooth Flow Fields. *Comput. Vis. Image Underst.*, 63(1):75–104, 1996.

[2] T. M. Breuel. Fast Recognition using Adaptive Subdivisions of Transformation Space. In *CVPR 92*, pages 445–51, 1992.

[3] T. M. Breuel. On the Use of Interval Arithmetic in Geometric Branch and Bound Algorithms. *Pattern Recogn. Lett.*, 24(9-10):1375–1384, 2003.

[4] D. Cremers and S. Soatto. Motion Competition: A Variational Approach to Piecewise Parametric Motion Segmentation. *Int. J. Comput. Vision*, 62(3):249–265, 2005.

[5] M. A. Fischler and R. C. Bolles. Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. *Communications of the ACM*, 24(6):381–395, 1981.

[6] D. A. Forsyth and J. Ponce. *Computer Vision: A Modern Approach*. Prentice Hall Professional Technical Reference, 2002.

[7] D. Geman. Stochastic Model for Boundary Detection. *Image Vision Comput.*, 5(2):61–65, 1987.

[8] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, New York, NY, USA, 2003.

[9] M. Irani and P. Anandan. About Direct Methods. In *ICCV '99: Intern. Workshop on Vision Algorithms*, pages 267–277, London, UK, 2000.

[10] A. Jepson and M. J. Black. Mixture Models for Optical Flow Computation. In *CVPR 93*, pages 760–761, 1993.

[11] M. J. Jones and J. M. Rehg. Statistical Color Models with Application to Skin Detection. *International Journal of Computer Vision*, 46(1):81–96, 2002.

[12] G. Kühne. *Motion-based Segmentation and Classification of Video Objects*. PhD thesis, University of Mannheim, 2002.

[13] V. Lepetit and P. Fua. Monocular Model-Based 3D Tracking of Rigid Objects: A Survey. *Foundations and Trends in Computer Graphics and Vision*, 1(1), 2005.

[14] D. W. Murray and B. F. Buxton. Scene Segmentation from Visual Motion using Global Optimization. *IEEE Trans. Pattern Anal. Mach. Intell.*, 9(2):220–228, 1987.

[15] T. Schoenemann and D. Cremers. Near Real-time Motion Segmentation using Graph Cuts. In *Pattern Recognition (Proc. DAGM)*, volume 4174, pages 455–464, Berlin, Germany, 2006.

[16] A. Smolic. *Globale Bewegungsbeschreibung und Video Mosaiking unter Verwendung parametrischer 2-D Modelle, Schätzverfahren und Anwendungen.* PhD thesis, RWTH Aachen, 2001.

[17] C. Tomasi and T. Kanade. Detection and Tracking of Point Features. Technical Report CMU-CS-91-132, CMU, 1991.

[18] A. M. Tourapis. Enhanced Predictive Zonal Search for Single and Multiple Frame Motion Estimation. In *Proc. SPIE Conf. Visual Communications and Image Processing*, pages 1069–1079, Lugano, Switzerland, 2002.

[19] J. Y. A. Wang and E. H. Adelson. Layered Representation for Motion Analysis. In *CVPR 93*, pages 361–366, 1993.

[20] Y. Weiss. Smoothness in Layers: Motion Segmentation using Nonparametric Mixture Estimation. In *CVPR 97*, pages 520–526, San Juan, Puerto Rico, 1997.