GIBBS RANDOM FIELD MODEL BASED 3-D MOTION ESTIMATION
BY WEAKENED RIGIDITY

A. Aydin Alatan and Levent Onural

Electrical-Electronics Engineering Department
Bilkent University
06533, Bilkent Ankara
e-mail: alatan@ee.bilkent.edu.tr ve onural@ee.bilkent.edu.tr

ABSTRACT

3-D motion estimation from a video sequence remains as a challenging problem. Modelling of local interactions between 3-D motion parameters is possible by using Gibbs Random Fields. An energy function which gives the joint probability distribution of motion vectors, is constructed. Most probable motion vector set is found by maximizing the probability, represented by this distribution. Since 3-D motion estimation problem is ill-posed, the regularization is achieved by an initial rigidity assumption. Afterwards, rigidity is weakened hierarchically, until the finest level is reached. At the finest level, each point has its own motion vector and the "weak-connection" between these vectors are described by the energy function. The high computational cost is decreased considerably by the multiprecision approach. The simulation results support all our discussions.

1. INTRODUCTION

The existing 3-D motion estimation methods have some drawbacks. These drawbacks can be summarized as:

- Susceptibility to noise [1]
- Structural constraints, like rigidity, planarity [1-5]
- Errors due to discrete differentiation [2]
- Lack of segmentation of the scene [1],[2],[4],[5]
- Orthographic projection [4],[5]

We propose a new approach to the 3-D motion estimation problem, which attempts to eliminate the drawbacks explained above. Our basic idea is to formulate the problem in such a way that, all the a priori information about the motion can be inserted into a cost function. Similar approaches were used successfully in image segmentation [6],[7] and restoration [8], and also 2-D optical flow determination [9],[10]. The cost function is also equivalent to the energy function of a Gibbs distribution, which is written by defining some local interactions between its neighbors. These local interactions permit looser relations between neighboring parameters, in contrast to the rigidity assumption, which is ultimately tight. Therefore, our approach may achieve non-rigid, or weakly-rigid, motion estimation, as well as rigid body motion estimation.

Furthermore, in our approach, the segmentation of moving objects can be achieved simultaneously with estimation of motion parameters. This is achieved, however, by adding new variables to the cost function. By increasing the number of variables, the cost function becomes difficult to minimize. The motion parameters are defined at each point on the object. Non-uniqueness of motion at every point and adding some extra fields for segmentation, makes the the cost function non-convex. The computation time for getting the global minimum becomes excessively high. At that point, a hierarchical approach solves this problem. Initially, the object can be assumed to be rigid and afterwards the rigidity can be weakened hierarchically, by decreasing the size of the rigid portions of the object. After each level, the solution at one hierarchy will serve as an initial estimate for the next one. Such an approach can be successful for avoiding local minima, which are expected from our cost function.

2. MODELLING 3-D MOTION ESTIMATION PROBLEM

2.1. Structural Model

3-D motion of a point on an object can be defined as below:

\[
\begin{bmatrix}
X' \\
Y' \\
Z'
\end{bmatrix} =
\begin{bmatrix}
1 & w_z & -w_y \\
-w_z & 1 & w_x \\
w_y & -w_x & 1
\end{bmatrix}
\begin{bmatrix}
X \\
Y \\
Z
\end{bmatrix} +
\begin{bmatrix}
T_x \\
T_y \\
T_z
\end{bmatrix}
\]

In the equation above, \(w_x, w_y, w_z\), are the rotation angles around the axes, \(x, y, z\) respectively. The above equation is valid for small values of the rotation angles. The \(T_x, T_y, T_z\) are the translation parameters of the object along the axes. According to the equation, the point \((X, Y, Z)\) in the 3-D space, moved to the point
\((X', Y', Z')\) after making rotations and translations. Although, this model enables using only one motion parameter set for a rigid object, we define this motion at each point on the object separately. The projection of the object coordinates into the image plane, is achieved by perspective projection, which is

\[
x(t) = \frac{X(t)}{Z(X,Y,t)}, \quad y(t) = \frac{Y(t)}{Z(X,Y,t)}
\]

where \(x(t)\) and \(y(t)\) are the image plane coordinates of the 3-D point \((X, Y, Z)\). After perspective projection of the points in 3-D space onto the image plane, the displacement of the points in the image can be written in terms of image plane coordinates, 3-D motion parameters and depth as:

\[
x'(t + \Delta t) = \frac{x(t) + y(t)w_x - w_y + (T_x/Z(t))}{1 + x(t)w_y - y(t)w_x + (T_z/Z(t))}
\]

\[
y'(t + \Delta t) = \frac{y(t) - x(t)w_x + w_y + (T_y/Z(t))}{1 + x(t)w_y - y(t)w_x + (T_z/Z(t))}
\]

In this equation, \((x, y)\) is the coordinate of a point on the image at time \(t\), moving to another coordinate \((x', y')\) at time \(t + \Delta t\), after 3-D motion. It should be noted that in this work, the depth information, \(Z(x, y)\), is assumed to be known.

Correct motion parameters must match the same point of an object on two consecutive frames, by using the intensity values, which can be shown as

\[
I_t(x, y) = I_{t+\Delta t}(x', y')
\]

The coordinates \((x, y)\) and \((x', y')\) are related with Equation 3. \(I_t\) and \(I_{t+\Delta t}\) are the “ideal” intensity values, which are projected from the 3-D environment into the image plane without considering illumination and intensity changes between frames and noise in the imaging environment. However, the observed intensities, \(I_t\), might differ from the ideal ones, \(I_t\), depending on the scene and hence Equation 4 may not hold.

2.2. Observation Model

\(I_t\) and \(I_{t+\Delta t}\) may differ due to 3 main reasons:
1. Existing noise in the scene
2. Illumination changes in the environment
3. Intensity changes on the specular surface of the moving object

In our model, only noise is taken into consideration. Assuming \(I_t\) and \(I_{t+\Delta t}\) are corrupted by Gaussian noise, the joint distribution of the displaced pixel differences must be jointly Gaussian, whose exponent is given by

\[
\frac{-1}{2\sigma^2}\sum_{x,y}(I_t(x,y) - I_{t+\Delta t}(x', y'))^2
\]

The \(\sigma\) can be adjusted according to the SNR of the image.

The other two reasons of observing different intensities on the image, can be also taken into account by detecting the direction of illumination and calculating surface normal directions, and properly adding a new term to Equation 5. This part is left as a future work.

2.3. 3-D Motion Vector Model

The motion parameters can be modelled as random variables. \(\mathbf{w}(x, y)\) is a vector random variable and it is defined on the lattice \(\Lambda_m\) (Figure 2(a)), on which image intensities \(I(x, y)\) are also defined. Each element of \(\mathbf{w}(x, y)\) can take values from a finite set, which also determines the range and the resolution of the motion parameters.

The joint probability distribution of \(\mathbf{w}\) can be written in terms of a Gibbs distribution, which is generally defined as

\[
P(W = w) = \frac{1}{K} e^{-U(w)/T}
\]

\[
U(w) = \sum_c V_c(w)
\]

In the above equation, the joint probability is given as an exponential distribution, where the energy (or cost) function, \(U(w)\) is the sum of some clique functions, \(V_c(w)\). The clique functions are defined on cliques, which are simply the sets of combinations of neighboring pixels \(B\). These clique functions represent the interactions between neighboring variables and reflects the a priori knowledge to the distribution. \(K\) is simply a normalization constant.

There are two important properties of the motion vector fields and these properties must determine the clique and hence the energy function. For many of the practical scenes, the 3-D motion parameters of local object points are similar (not necessarily to be equal). The discontinuity of the 3-D motion parameters must be modelled as well as the continuity, since it is probable to observe multiple objects in the scene, moving with different motion parameters, passing each other and generating motion boundaries. In order to be able to segment objects, this second property needs some extra fields, apart from motion and intensity fields. As an extra point, it should be noted that, some unpredictable and model failure points must be discriminated from the others. The points on the uncovered background of a moving object are useless for the estimation of motion and can be assumed as unpredictable points. Similarly, the 3-D motion of some object points cannot be modelled by our motion definition and therefore must be dropped during estimating. This property also needs a new field to be modelled.

Taking into account all the ideas above, an energy function is proposed for the estimation of 3-D motion.
parameters from consecutive two frames and depth field. More explicit version of this energy function can be found in the Appendix.

\[
U(\bar{w}, I, s \mid I_i, I_{i+\Delta t}, Z(x,y)) = U_m(I_i, I_{i+\Delta t}, s) + \lambda_m U_m(\bar{w}, I) + \lambda_s U_s(s, I) + \lambda_l U_l(I, I_i) + U_f(Z, \bar{w})
\]

The first term, \( U_m \), in the energy function achieves matching of intensities between consecutive frames, \( I_i \) and \( I_{i+\Delta t} \) and it is equal to the Equation 5. However this distribution is valid for only the predictable points in the image, which are marked by the binary random field, \( s(x,y) \), defined on \( \Lambda_m \). The points, for which \( s(x,y) \) equals to one, is assumed to be unpredictable or model failure. The second term, \( U_m \), stands for the continuity of the neighboring motion parameters. It is defined on some cliques, \( e(x,y) \), which are basically four neighbors of the point \((x,y)\) on the lattice \( \Lambda_m \) (see Appendix). For the expected motion fields, \( L_2 \) norm of the neighbor motion vectors must be as small as possible. \( l(x,y) \) is a binary random field and prevents getting high penalties along motion boundaries. \( l(x,y) \) field is defined on a lattice, \( \Lambda_l \), which is dual of \( \Lambda_m \) and both of them are shown in Figure 1(a). The next term, \( U_s \), supports the idea that unpredictable or model failure points, must be in groups in the image.

If a penalty is not given for setting \((x,y)\) to one, then all values on this discontinuity field must be one, which will make the second term zero for all values of \( \bar{w}(x,y) \) (see Appendix). These penalties must depend on the expected local structure of the discontinuity field [8] and shown in Figure 1(b). \( U_m \) term also uses spatial discontinuity information, in order to locate the motion discontinuities. It encourages the locations for which \( l(x,y) \) equals to one and spatial discontinuity is high. The very last term, \( U_f \) gives a high penalty for the motion vectors which points a coordinate in 3-D space, which is occupied by another object point. Details about the energy function can be found in [14].

### 3. MULTiresolution EXTENSIONS

#### 3.1. Hierarchical Rigidity

The uniqueness of the 3-D motion parameters is investigated in the literature [1],[4],[5]. When 3-D motion vectors associated with the points of an object can be different from the neighboring vectors, the uniqueness of the motion is no more guaranteed. However, uniqueness is necessary for the convergence of the algorithm. In order to have a unique solution a constraint, which makes the neighboring pixels exactly equal to each other at some predefined neighborhood, can be defined. Instead of defining the motion vectors at each point on the image, they can be defined, over subsampled versions of the lattice \( \Lambda_m \), as shown in Figure 2. This will have an effective result of assigning the same motion vector to some number of image points, which are defined on the finest level of hierarchy. Therefore, an image can be assumed to be consisting of some rigid rectangular patches with different sizes at each level. In the coarsest level (Level 0), there are large rigid bodies. Whereas going down through the levels, we finally reach to the finest level, on which every image point has its own motion vector associated with it. At each level of hierarchy, the interaction between neighboring motion vector block still exist. At each level, the algorithm creates some motion vector sets, which will serve as an initial estimate for the next level, which can be shown as:

\[
\Delta \bar{w}^k = \bar{w}^{k-1} + \Delta \bar{w}^k
\]

where \( \Delta \bar{w}^k \) is tried to be estimated at level \( k \).

The advantage of such an approach is to guarantee the uniqueness of the motion parameters at the coarsest level (easy convergence). However, this guarantee is valid by the assumption that the predefined neighborhood is rigid. The results are propagated through levels, serving as initial estimate for the search algorithms in the next levels. At the finest level, we still reach to our initial "weakly rigid" model, in which the motion vectors are weakly connected by a similarity term in the energy function.

#### 3.2 Multiprecision for Motion Parameters

There are six 3-D motion parameters for each point on the image. According to the precision of each parameter, the 6-D search space may become enormous. However, during a search operation, beginning from a coarse precision of the motion parameters, the parameters can be refined in a hierarchical way, similar to hierarchical block matching motion estimation [11]. This multiprecision can be shown as:

\[
\Delta \bar{w}^k_m = \bar{w}^{k}_{m-1} + \Delta \bar{w}^k_m
\]

where \( \Delta \bar{w}^k_m \) is the precision detail, which is estimated at precision level \( m \). Such an approach decreases the computation time considerably, although it does not achieve an exhaustive search.

### 4. MINIMIZATION OF THE ENERGY FUNCTION

The energy function must be minimized in order to find the maximum a posteriori estimate of the \( \bar{w}(\ldots), l(\ldots) \) and \( s(\ldots) \) fields. There are excessive number of unknowns and there exist methods to solve such optimization problems. Simulated Annealing [12] (SA)
is one of the most popular among them. It is computationally very costly, and converges in infinite time. However, in finite time and with some suboptimal temperature schedules, it gives reasonable results. Iterated Conditional Modes [13] (ICM) is another popular optimization method, which has "hard decision" nature in contrast to annealing methods. It maximizes the marginal distribution in localities. It converges to comparable results with annealing based methods, if the initial estimates are successful. Both SA and ICM are used during simulations, for the minimization of the energy function in an hierarchical manner.

5. SIMULATION RESULTS

Although, the formulation is valid for non-rigid motion, all the simulations are carried on artificial sequences, which have rigid motion (Figure 3 and Figure 5). The experiments consist of four stages: 1. Validation of hierarchical rigidity 2. Comparison between single and multiprecision 3. Segmentation for multi moving objects 4. Noise analysis

In the first step, ICM is used in the coarsest (8x8 blocks) level. The results are used as initial estimates for SA, which is used in the finer levels. A typical result is shown in Figure 4. The estimated parameter w_i is shown by intensities over the image (a) and a histogram (b) for the finest level (1x1 blocks).

The values of the constants, λ_i, in the energy function is found by experimentation and tabulated in Table 1.

<table>
<thead>
<tr>
<th>λ_mat</th>
<th>λ_s</th>
<th>λ_i</th>
<th>λ_disc</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>5</td>
<td>0.5</td>
<td>20</td>
</tr>
</tbody>
</table>

Table 1. λ values

In the second stage, the precision of motion parameters are decreased by two and hierarchical rigidity is applied. The precision of resulting coarse values are increased by applying the same algorithm with a finer precision around the initial estimates, according to Equation 8. In Table 2, hierarchical rigidity with full precision is compared to multiprecision with 2-levels.

<table>
<thead>
<tr>
<th>r_w</th>
<th>r_u</th>
<th>r_w</th>
<th>r_u</th>
<th>r_w</th>
<th>r_u</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.81</td>
<td>0.88</td>
<td>0.82</td>
<td>0.88</td>
<td>0.96</td>
<td>0.90</td>
</tr>
</tbody>
</table>

Table 2. Single vs. Multi precision

In this table, the ratio values for each motion parameter is equal to

\[ r_a = \frac{||d_{actual} - a_{single}||}{||d_{actual} - a_{multi}||} \]

(9)

where a is one of the motion parameters and the norm is evaluated at each point on the image. The deterioration between single and multiprecision is in acceptable limits. It should be noted that, the computational saving for using a 2-level multiprecision is about 32 times for six 3-D motion parameters.

In the third step, a scene with 2 cubes moving with different speeds are examined by hierarchical rigidity (Figure 5). The results in Figure 6 show that, the method can easily achieve segmentation of the cubes and assign their 3-D values correctly.

As a final step, a noise analysis is performed on our proposed method. The images are corrupted by Gaussian noise, obtaining frames, having SNR_recalc values as 28dB and 43dB. It is observed in Figure 7 that, above 30dB the method performs acceptable.

6. CONCLUSIONS

Hierarchical rigidity is found to be necessary in estimating 3-D motion vectors, modelled by Gibbs distribution. By forcing some number of points to use the same motion vector parameter, convergence to a more rigid solution becomes easier. On the other hand, multiprecision 3-D motion vector approach reduces the computational complexity by more than an order of magnitude and there is a minor degradation in the recovery of the motion vectors by this approach. The proposed hierarchical approach solves the segmentation of multi-moving objects, by having a dense 3-D motion field and using some extra segmentation variables in the energy function. The experiments also showed that the proposed method is relatively robust in the presence of noise.

Currently, only drawback seems to be the computation time. The estimation of depth is left as a future work. Although, only small number of experiments are achieved, it can be stated that, our method is a promising approach to non-rigid motion estimation.

7. REFERENCES

[1] J. Weng, N. Ahuja and T.S. Huang, "Optimal Motion and Struc-
[2] B.K.P. Horn and E.J. Weldon, "Direct methods for recovering mo-
graphic projection", IEEE Trans. on PAMI, vol. 11, no. 5, pp. 536-
APPENDIX

\[ U_o(l_i, l_{i+1}, x) = \sum_{x} \sum_{y} (1 - \alpha(x, y)(l_i(x, y) - l_{i+1}(x', y')))^2 + \beta(x, y)T_x \]

\[ U_m(u_i, t) = \sum_{x} \sum_{y} \sum_{l(x, y)} \|u(x, y) - u(x, y, z)\|^2 (1 - \lambda(x, y)) \]

\[ U_s(s, l) = \sum_{x} \sum_{y} \sum_{l(x, y)} \phi(s(x, y), s(x, y, z))(1 - \lambda(x, y)) \]

\[ U(\lambda_i, \lambda) = \sum_{x} \sum_{y} L(x, y) \]

\[ + \lambda \sum_{x} \sum_{y} \sum_{l(x, y)} \frac{\lambda(x, y)}{1 + (l(x, y) - l(x, y, z))^2} \]

\[ U_2(Z, \lambda) = \theta(x, y, Z, u(x, y)) \]

---

Figure 1: (a) ‘o’ : \( \lambda_m \), \( \lambda \): \( \Lambda \), (b) Penalties of line field

---

Figure 3: 2 original frames of “Cube” sequence

Figure 4: (a) \( w_1 \) (intensity rep.) (b) \( w_2 \) (histogram rep.)

Figure 5: 2 original frames of “Cubes” sequence

Figure 6: Histogram of \( t_y \) parameter for “Cubes”

Figure 7: Noise analysis of “Cube” sequence