Abstract—This paper presents a block-overlap-based method for handling the ill-posed nature of motion estimation. The proposed method uses the volume of motion-compensated block overlap as an additional term in minimizing the overall energy. By reducing the amount of block overlap, the proposed method results in a significant improvement in the quality of the motion field. Experimental results show that the proposed method outperforms several existing methods in the literature in terms of motion vector (MV) and interpolation quality. In terms of interpolation quality, our algorithm outperforms all other block-based methods as well as several complex optical flow methods. In addition, it is the fastest non-GPU implementation at the time of this writing.

I. INTRODUCTION

Motion estimation has become a defining factor in applications such as object tracking, frame-rate conversion, super-resolution, and 3D TV. In such applications, it is necessary to estimate the true motion of objects. Finding the true motion, however, is an ill-posed problem because of the possibility of multiple minima. To reduce the number of minima, i.e., to convert the motion estimation problem into a convex problem, regularization constraints are often introduced. However, even with regularization constraints, state-of-the-art motion estimation algorithms perform poorly in untextured regions, occluded regions, and for small structures [1].

Since many applications require a real-time approach to motion estimation, we focus on block-matching-based approaches. Block matching makes use of the translational-motion model and brightness-constancy assumption to estimate the motion of blocks between image pairs. To determine the best matching block, correlation-based approaches are generally favored due to their robustness and low complexity [2]. Block matching methods in the literature almost exclusively use the Sum of Absolute Deviations (SAD) correlation metric, which computes the $\ell_1$ norm of pixel differences.

To regularize the results of block matching, low-complexity smoothness constraints have been introduced [3][4]. These constraints limit the solution based on the assumption that the motion field should be locally constant. The effects of different smoothness constraints on the quality of the motion field were evaluated in [3][4].

Although regularizing the motion estimation problem using smoothness constraints reduces the number of possible solutions, it does not force a unique solution. In fact, unless the regularization function is strictly convex, multiple solutions may exist. In previous approaches [5][3][4], the chosen solution depends on the block matching search order and on the order in which the smoothness constraints are applied. In this paper, we propose a block-overlap-based minimization approach for both block matching and smoothness to reduce the order dependence and improve the quality of the motion field.

In the next sections, we examine the deficiencies of some existing block-based methods and improve upon these deficiencies by minimizing the amount of block overlap. In section II, we develop the energy minimization framework and discuss the shortcomings of previous work. The block-overlap-based minimization framework is introduced in section III. Experimental results are given in section IV, and the conclusions of the paper are presented in section V.

II. PREVIOUS WORK

In this section, we first develop the energy minimization framework that will be used in subsequent sections. With this framework in place, we then discuss the shortcomings of previous work with regard to the choice of the optimal motion vector (MV).

A. Motion Estimation Framework

The motion estimation problem can be formulated as the following energy minimization problem:

$$E = \min \{ D(I_0, I_1, v_i) + \lambda R(v_i) \} , \quad (1)$$

where $D(I_0, I_1, v_i)$ is a data term that measures the similarity of blocks in images $I_0$ and $I_1$ for a given MV $v_i$, $R(v_i)$ is a regularization term which penalizes deviations in the smoothness of the motion field, and $\lambda$ is used to weight the regularization term over the data term. The goal of the motion estimation problem is to choose a MV $v_i$ such that the energy in (1) is minimized.

The majority of the block matching methods in the literature use the SAD error metric for the data term in (1) because of its robustness and low-complexity [6]. The SAD error is given as follows:

$$D(I_0, I_1, v_i) = \sum_{x \in B} |I_0(x) - I_1(x + v_i)| , \quad (2)$$

where $x$ is the pixel position in a block $B$ of pixels.
Robust potential functions such as the $\ell_1$ and $\ell_2$ norms [7] are often used for the regularization term [3][4]. In the work of [3], Bartels and de Haan analyzed several different potential functions and candidate sets. A candidate set refers to the spatial and/or temporal MVs of neighboring blocks. The authors found that quality of the motion field was not highly sensitive to the number of MVs in the candidate set. In previous work, we found that using eight spatial candidates provided the best quality motion field [8]. Similar to Bartels and de Haan, we chose the optimal potential function as follows:

$$\mathcal{R}(v_i) = \sum_{j \in C^s} \|v_i - v_j\|_1, \tag{3}$$

where $v_i$ and $v_j$ are spatial MVs in candidate set $C^s$ and $v_i \neq v_j$. An example candidate set of eight-connected spatial MVs is shown in Fig. 1. With respect to (3), the best MV for the center block in Fig. 1 is the MV that is most similar to its spatial neighbors. Therefore, replacing candidate $C_5$ with one of its neighbors would result in the minimum value of $\mathcal{R}(v_i)$. However, this is only half of the picture; the chosen MV $v_i$ must minimize both the data and regularization term in (1) in order to be considered the optimal MV.

B. Shortcomings of previous work

The goal of any block matching algorithm is to determine which block in the adjacent frame minimizes the SAD error in (2). The failure to locate the best block is generally due to brightness variations, multiple matches, dependence on search order, complex object motion (e.g., deformations), and occlusions. Brightness variations, complex motion, and occlusions are limitations of the block-based model and the inability to estimate disappearing objects. On the other hand, for multiple matches and search-order dependence, an optimal block often exists. To find the optimal block, we must choose an effective search order, and more importantly, choose between blocks with the same SAD errors.

It was shown in [9] that a spiral search order outperforms the typical raster scan order that was proposed in early block matching algorithms. Spiral search is a natural extension of any hierarchical block matching algorithm since the previous level of the hierarchy provides an initial estimate of the motion, i.e., the best matching block at a subsequent level is likely to be in the vicinity of the initial estimate. Even though spiral search helps to reduce the dependence on search order, multiple matches may still exist. As far as we know, no other block matching methods attempt to discriminate between similar matches. Instead, the first searched block with the minimum SAD (or other correlation) error is chosen. In section III, we show that block overlap minimization can be used to choose the best matching block among blocks with the same SAD errors.

Following the block matching step, smoothness constraints are applied to find the MV that minimizes (1). However, since neither of the terms in (1) is strictly convex, there will not necessarily be a unique minimum. Therefore, it is necessary to choose between MVs which produce the same overall energy. Without an explicit way to discriminate between MVs that produce the same overall energy, the chosen MV will depend on the order in which the spatial candidates $C^s$ are tested in (3). Fortunately, as we will show in section III, block overlap minimization is also suited to handle such cases.

III. BLOCK OVERLAP MINIMIZATION

The block overlap approach in this section was first introduced by the authors in the context of MV validity and hybrid de-interlacing [10]. In [10], it was shown that the block overlap validity method significantly outperforms other validity methods in the literature. In this section, we give an overview of the block overlap method and extend it to be used as an error metric for improving the quality of the motion field.

A. Method Overview

To motivate the block overlap minimization approach, we begin by considering a pair of images whose motion we wish to estimate. Let the current image $I_1$ be divided into a grid of square blocks of size $BS$, and let the grid of the adjacent image $I_0$ be determined by the motion-compensated (MC) blocks, as shown in Fig. 2. The position of a block in $I_1$ is denoted as $y$ and the position of the MC block in $I_0$ as $z = y + v$, where $v$ is the MV. To simplify the analysis, we assume a square grid; however, the analysis also holds for non-square grids.

In the ideal case, there exists an injective function $f : I_1 \rightarrow I_0$ such that each block in $I_1$ is mapped to a unique block in $I_0$. However, for real images with various motion types, the process of mapping blocks from $I_1$ to $I_0$ is non-surjective, i.e., blocks in $I_1$ may be mapped to same block position in $I_0$, and the whole of $I_0$ is not necessarily filled. An example of the
block mapping for real images is shown in Fig. 3. As shown in Fig. 3, the mapped MC blocks for real images may overlap each other to different degrees. We wish to characterize the degree of overlap as the uncertainty in the motion estimation decision, i.e., the error for any given MV will depend on the amount to which its MC block overlaps with other MC blocks.

An MC block \( z \) will cover an area of \( B_S^2 \) in \( I_0 \) if no overlap occurs. However, when blocks in \( I_0 \) overlap, a volume (perhaps nonuniform) is generated, which we denote as \( L_y(z) \). Larger values of \( L_y(z) \) indicate less confidence in the MV. The algorithm used to determine \( L_y(z) \) is given in Algorithm 1, where \( z_i \) and \( z_j \) represent the vertical and horizontal position of block \( z \), respectively.

Algorithm 1: Calculate volume for each \( L_y(z) \)

```plaintext
For all blocks in \( I_1 \), set \( z = y + v \)
for \( k = z_i, k < z_i + B_S, k++ \) do
  for \( l = z_j, l < z_j + B_S, l++ \) do
    \( L_y(k, l) += 1 \)
```

B. Block overlap minimization for motion estimation

The volume \( L_y(z) \) introduces an additional error metric for determining which MV minimizes the overall energy. Therefore, we modify the energy expression of (1) as follows:

\[
E = \min \left\{ D^1(I_0, I_1, v_1) + \lambda R(v_1) + \mathcal{O}_2(I_0, I_1, v_1) \right\},
\]

where \( \mathcal{O}_2(I_0, I_1, v_1) \) represents the overlap volume of the MC block in \( I_0 \) given by \( v_1 \), i.e., \( L_y(z) \) in Algorithm 1, and the modified data term, \( D^1(I_0, I_1, v_1) \), is given as

\[
D^1(I_0, I_1, v_1) = \sum_{x \in B} |I_0(x) - I_1(x + v_1)| + \mathcal{O}_1(I_0, I_1, v_1).
\]

Note that although \( \mathcal{O}_1(I_0, I_1, v_1) \) and \( \mathcal{O}_2(I_0, I_1, v_1) \) both represent block overlap volumes, we use the subscripts to denote the order in which they are applied.

As discussed in section II-B, traditional block matching chooses the first searched block with the minimum SAD error. However, the optimal block may be another block with the same SAD error. Therefore, blocks with the same SAD error should be tested in (5) to determine which minimizes the combined error.

Once the best MV is chosen for all image blocks with respect to (5), the new MVs are then tested in (4). If the tested MVs produce the same value for the sum of \( D^1(I_0, I_1, v_1) + \lambda R(v_1) \), the chosen MV will depend on the order in which the MVs are tested. In the proposed algorithm, however, the \( \mathcal{O}_2(I_0, I_1, v_1) \) term is used to further discriminate between MVs where the sum results in the same value. Note that the MV \( v_i \) used in \( \mathcal{O}_1(I_0, I_1, v_i) \) and \( \mathcal{O}_2(I_0, I_1, v_i) \) is in general not the same MV since \( v_i \) in (4) may come from a neighboring MV in \( I_1 \).

C. Refinement

To summarize the results of the previous sections, the optimal MV is found by 1) determining the minimum SAD, 2) discriminating among the same (minimum) SAD errors by minimizing the overlap, 3) jointly applying the SAD and smoothness constraints, and 4) using overlap again to discriminate among MVs which produce the same combined SAD and smoothness.

To understand why our approach further benefits from the refinement discussed in this section, we first examine the order in which steps 1)-4) are applied. Before steps 3) and 4) can be applied, it is first necessary to evaluate steps 1) and 2) for every block in the image. Otherwise, it is not possible to use an eight-connected neighborhood in the smoothness constraints of (3). Since steps 1) and 2) are evaluated in a raster scan order, the block overlap is minimized only with respect to blocks in \( I_1 \) whose overlap has already been determined, i.e., the overlap has not be determined for all neighboring blocks. Therefore, we propose a refinement in order to further reduce the block overlap once the initial overlap for all blocks has been determined.

Rather than reiterating steps 1)-4) to further minimize the overlap, we propose a more efficient approach. To keep computational complexity low, we only apply refinement to those MVs whose blocks overlap with other MC blocks. The refinement algorithm is given in Algorithm 2.

Algorithm 2: Refinement Algorithm

```plaintext
For all pixel positions \((i, j)\) corresponding to blocks in image \( I_1 \) and their MVs \( v = \{v_x, v_y\} \),
\( z_i = i + v_y; z_j = j + v_x \)
if \( (L_y(z_i, z_j) > B_S^2) \) then
  for \( k = z_i - B_S/2; k < z_i + B_S/2; k++ \) do
    for \( l = z_j - B_S/2; l < z_j + B_S/2; l++ \) do
      \((k, l) = \arg \min_k \text{SAD}(I_0(i, j), I_1(k, l)) + L_y(k, l)\)
      \( v_y = k - i; v_x = l - j \)
```

As shown in Algorithm 2, only MC blocks with an overlap volume greater than \( B_S^2 \) are processed in the refinement step, which greatly reduces the number of tested blocks. The MC blocks are shifted in both the horizontal and vertical directions by \( B_S/2 \) in order to determine if neighboring blocks in \( I_0 \) result in less overlap. The shift factor of \( B_S/2 \) was chosen to provide a good trade-off between quality and performance.
Note that although we include the SAD term in Algorithm 2 for clarity, only blocks in $I_0$ with the same or smaller SAD errors are used in Algorithm 2.

The overall progression for the proposed algorithm (steps 1)-4) and refinement) is given in Algorithm 3.

**Algorithm 3 : Proposed Motion Estimation Algorithm**

1: Form image hierarchy and begin at lowest-resolution level.
2: For all blocks in image $I_1$, find blocks in image $I_0$ with lowest SAD error using spiral search.
3: For blocks with the same SAD error, choose the block with the smallest overlap.
4: Minimize the SAD of the chosen block together with the eight-connected spatial neighbors, $C^8$.
5: For MVs that results in the same energy in Line 4, choose the block (MV) with smallest overlap.
6: Iterate Line 4 until MVs converge, reduce the block size, and repeat.
7: Pass converged MVs to next level of hierarchy and return to Line 2. Repeat until highest-resolution level of hierarchy is reached.

**IV. RESULTS**

In all the results that follow, we used the method given in Algorithm 3 to obtain quarter-pixel MVs. A three-level hierarchy was used, but we avoided interpolation by using a modified version of [11]. For images with a VGA resolution, the run time of our method is approximately two seconds using unoptimized code.

To show the concept behind block overlap minimization, we compare the motion estimation results with and without block overlap minimization. We illustrate the MV quality improvement of block overlap minimization for one of the ground truth sequences from [12]. The computed MVs using (4), (1), and the ground truth MVs are shown in Fig. 4. As shown in Fig. 4, the MVs from overlap minimization (Fig. 4(a)) more closely resemble the ground truth MVs of Fig. 4(c).

We note that the blue background does not change between images; however, applying (1) produces incorrect MVs in this region (Fig. 4(b)). In addition, there are missing MVs toward the bottom of Fig. 4(b). In terms of endpoint error, overlap minimization (4) results in a 1.9 dB improvement over the energy of (1).

To further demonstrate the effect of overlap minimization, we show a comparison between the MC blocks of (4) and (1) in Fig. 5. As shown in Fig. 5, the distribution of MC blocks in Fig. 5(a) is more uniform than that of Fig. 5(b).

We also demonstrate the effectiveness of overlap minimization using the eight ground truth test sequences from Middlebury University [13]. In Table I, we show comparisons of endpoint error for the MVs of (4) and (1). As shown in Table I, block overlap minimization results in an improvement of 0.33 dB when averaged over all of the sequences. For the sequences in Table I where no improvement was reported, the energy function of (1) does a sufficient job of minimizing the overlap through the use of the SAD and smoothness constraints, i.e., it is not necessary to incorporate an overlap term. For the "Grove 2" sequence, the large improvement can be attributed to the ability of the overlap method to reduce errors around the occluded edges of the leaves. The visual improvement of the block overlap method (4) over that of (1) for a small region of the "Grove 2" sequence is shown using

<table>
<thead>
<tr>
<th>Sequence</th>
<th>Endpoint Error for (4)</th>
<th>Endpoint Error for (1)</th>
<th>Improv. in dB</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dimetrodon</td>
<td>0.217</td>
<td>0.217</td>
<td>0.00</td>
</tr>
<tr>
<td>Grove 2</td>
<td>0.214</td>
<td>0.338</td>
<td>1.99</td>
</tr>
<tr>
<td>Grove 3</td>
<td>0.691</td>
<td>0.712</td>
<td>0.13</td>
</tr>
<tr>
<td>Hydrangea</td>
<td>0.231</td>
<td>0.231</td>
<td>0.00</td>
</tr>
<tr>
<td>Rubber Whale</td>
<td>0.161</td>
<td>0.161</td>
<td>0.00</td>
</tr>
<tr>
<td>Urban 2</td>
<td>0.482</td>
<td>0.513</td>
<td>0.27</td>
</tr>
<tr>
<td>Urban 3</td>
<td>0.888</td>
<td>0.912</td>
<td>0.12</td>
</tr>
<tr>
<td>Venus</td>
<td>0.327</td>
<td>0.335</td>
<td>0.11</td>
</tr>
</tbody>
</table>

Table I, block overlap minimization results in an improvement of 0.33 dB when averaged over all of the sequences. For the sequences in Table I where no improvement was reported, the energy function of (1) does a sufficient job of minimizing the overlap through the use of the SAD and smoothness constraints, i.e., it is not necessary to incorporate an overlap term. For the “Grove 2” sequence, the large improvement can be attributed to the ability of the overlap method to reduce errors around the occluded edges of the leaves. The visual improvement of the block overlap method (4) over that of (1) for a small region of the “Grove 2” sequence is shown using

![Fig. 4. Visual comparison using MVs of (4), (1), and ground truth.](image-url)
Fig. 6. Screenshots taken from Middlebury benchmark [13] with our endpoint error results (top) and interpolation error results (bottom) highlighted.

In Fig. 7, improvements generated by the block overlap method can be seen in the top half of the images near the left edges of the leaves. Similar improvements throughout the “Grove 2” sequence are responsible for the large improvement reported in Table I.

In addition to the results shown in Table I, we also submitted our results to the Middlebury online database for comparison with other motion estimation algorithms. The online results are shown in Fig. 6. As shown in Fig. 6, our algorithm outperforms several others in terms of endpoint and interpolation error. For the interpolation error, our algorithm outperforms all other block-based methods as well as several complex optical flow methods with long run times, and it produces the smallest interpolation error for the “Evergreen” sequence compared to all 67 algorithms currently in the database. In addition, it is the fastest non-GPU implementation at the time of this writing.

V. CONCLUSION

The block-overlap-based minimization approach proposed in this paper uses the block overlap volume of MC blocks as an additional term in the energy function. By introducing the block overlap term, we were able to reduce the number of possible solutions in both block matching and the subsequent regularization of the motion field. It was shown in section IV that proposed method results in a more uniform distribution of blocks, which reduces also reduces the endpoint error. In addition, the published results for the Middlebury sequences show that our method performs well compared to other state-of-the-art methods in the literature, and it is well-suited for applications which require a real-time approach.
REFERENCES