A Novel Motion Field Anchoring Paradigm for Highly Scalable Wavelet-Based Video Coding

Dominic Rüfenacht, *Student Member, IEEE*, Reji Mathew, *Member, IEEE*, and David Taubman, *Fellow, IEEE*

***Abstract*— Existing video coders anchor motion fields at frames that are to be predicted. In this paper, we demonstrate how changing the anchoring of motion fields to reference frames has some important advantages over conventional anchoring. We work with piecewise-smooth motion fields, and use breakpoints to signal discontinuities at moving object boundaries. We show how discontinuity information can be used to resolve double mappings arising when motion is warped from reference to target frames. We present an analytical model that allows to determine weights for texture, motion, and breakpoints to guide the rate-allocation for scalable encoding. Compared with the conventional way of anchoring motion fields, the proposed scheme requires fewer bits for the coding of motion; furthermore, the reconstructed video frames contain fewer ghosting artefacts. The experimental results show the superior performance compared with the traditional anchoring, and demonstrate the high scalability attributes of the proposed method.**

***Index Terms*— Bidirectional hierarchical anchoring of motion fields, motion discontinuity modelling, wavelet-based scalable video coding, geometrically consistent prediction.**

I. Introduction

AT THE heart of any modern video coder is motion-compensated prediction, where the temporal correlation between frames is exploited to predict certain *target* frames from neighboring *reference* frames; only the prediction residual and side information (e.g., motion vectors) are encoded. From an implementational point of view, it seems a natural choice to anchor the motion field in the frame that is to be predicted (*target* frame). That is, each pixel in the target frame gets assigned a motion vector relating it to a reference frame. Somewhat counter-intuitively, we propose in this paper to anchor motion at reference frames; as we shall see, this fundamental change yields some major advantages over the conventional motion anchoring at target frames.

Almost every successful video coder employs block motion compensation (BMC) [1], where the target frame is partitioned into blocks, and each such *macroblock* gets assigned the "prediction" vector which results in the smallest prediction residual, or at least a good balance between prediction residual and coding cost. The reason we call this a *prediction* rather than a *motion* vector is because there is no attempt to model the underlying motion field. Over the past few decades, video codecs have evolved from using one fixed block size (H.261), to using a large variety of rectangular blocks in HEVC, the latest standardized codec [2]. The large variety of block sizes allows to better account for the fact that blocks have difficulty in representing motion in the vicinity of moving object boundaries, and to a certain extent *implicitly* model discontinuities in the motion field by having smaller block sizes in the vicinity of motion boundaries. For many scenes, the object motion is smooth, in which case the underlying motion field is piecewise smooth. HEVC allows for blocks to inherit the motion of their spatial neighbours, which favours smoothness within moving objects. Higher order motion models have been shown to be beneficial in scenes with background motion that is difficult to describe with blocks (e.g., rotation, zoom) [3]–[5].

Taking this approach further, it is reasonable to consider motion models which *explicitly* communicate motion boundaries, allowing for piecewise smooth motion fields that are able to follow the true scene geometry. In this paper, we show that such an approach can provide many advantages for video compression, especially for scalable video coders. The objective here is to eliminate artificial block boundaries, while efficiently describing true discontinuities in the motion flow. There is a body of research on estimating piecewise-smooth motion fields with sharp transitions at object boundaries [6], [7], but this remains a challenging task. As we will see, we require the discontinuities to be aligned across multiple motion fields originating from a given frame, which is a topic of ongoing research. In this paper, we work with ground truth motion fields, noting that results from the present paper will drive a motion estimation scheme appropriate for our framework. Quite apart from this, a significant obstacle to the use of such "optical flow" motion fields for video coding is the high communication cost. Recently, dense motion estimation methods for video coding have been proposed which optimise for both smoothness and compressibility of the motion field [8]. Young *et al.* [9] *explicitly* handle motion discontinuities and advocate *compression-regularized optical flow*, where piecewise-smooth motion fields are estimated, and discontinuities are explicitly coded. As shown in [10], such motion fields are highly *scalable*.

In video coding, *scalability* refers to an encoding of a video in an *embedded* way such that lower qualities

Manuscript received March 12, 2015; revised July 8, 2015 and September 14, 2015; accepted October 16, 2015. Date of publication October 30, 2015; date of current version November 18, 2015. This work was supported by the Australian Research Council under Grant DP14010427. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Yui-Lam Chan.

The authors are with the School of Electrical Engineering and Telecommunications, University of New South Wales, Sydney, NSW 2052, Australia (e-mail: d.ruefenacht@unsw.edu.au; reji.mathew@unsw.edu.au; d.taubman@unsw.edu.au).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIP.2015.2496332

(spatial, temporal, and bitrate) are embedded within higher qualities. Scalable video can be beneficial in a variety of applications. Examples include scenarios where the parameters of the decoding device are unknown during encoding (e.g., streaming of video over heterogeneous networks), or applications where *interactive navigation* within the video is beneficial (e.g., surveillance and medical applications). The latest standardized video codecs (H.264 and HEVC) have scalable extensions, H.264/SVC [11] and SHVC [12], respectively. The closed-loop nature of these codecs requires the base and enhancement layer(s) to be decided at the encoder, limiting their scalability attributes.

A natural alternative for *highly* scalable video coding involves the use of *wavelets*. Various *wavelet-based scalable video coders* (WSVC) have been proposed in the literature, which mainly differ in the order of application of the spatial and temporal *discrete wavelet transform* (DWT). Probably the most natural approach is *spatial domain motion compensated temporal filtering* ("t+2D"), where the wavelet transform is first applied in the temporal domain, followed by a 2D-DWT of the temporal subbands [13], [14]. Secker and Taubman [15] propose a lifting-based invertible motion adaptive transform (LIMAT), which uses a deformable mesh model for the motion which is able to model expansion and contraction of moving objects. Alternate structures have been proposed, including *in-band motion compensation* ("2D+t") schemes [16], as well as adaptive schemes [17] which adapt between the *"t+2D"* and *"2D+t"* models.

Even though there have been significant improvements in the field of scalable video coding, the rate-distortion performance of scalable video coders remains inferior to single-layer coding. One reason for this is the fact that scalable video coders have problems at *discontinuities* in the motion field [18]; band-limited sampling of motion fields smooths out the sharp transitions at moving object boundaries, which creates non-physical motion – a value interpolated between the motion of the foreground and the background object is in general a poor predictor. Wavelet-based schemes that handle motion discontinuities because of intrinsic properties of the setup have shown to have improved performance [19], [20]. For generic video data, a more general way of handling motion discontinuities is required. Mathew *et al.* [21] propose a highly scalable representation of discontinuities in both resolution and precision. Discontinuities are determined in a rate-distortion optimization framework, and signalled using *breakpoints*. These breakpoints are used to avoid wavelet bases from crossing discontinuity boundaries, and allow to efficiently code motion fields.

In the proposed method, discontinuity information allows the inference of auxiliary information (e.g., occluded regions), which has to be explicitly coded as side information in standard video codecs. More surprisingly, in combination with "true" motion fields, it allows to flip around the anchoring of motion fields, which proves to yield geometrically consistent predictions with quantized motion; in addition, significant savings in terms of motion field coding cost can be achieved. We refer to this scheme as *bidirectional hierarchical anchoring* (BIHA) of motion fields. While particularly useful

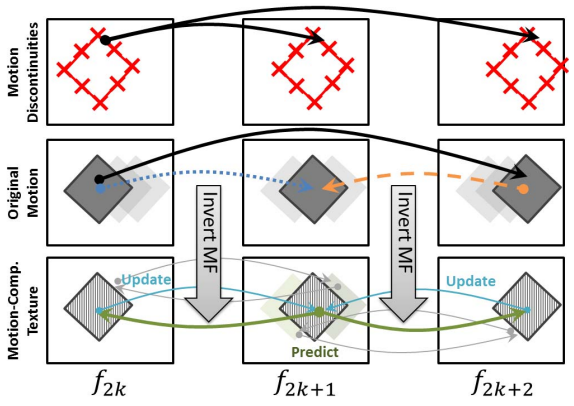


Fig. 1. Overview of the different steps involved in one level of proposed temporal transform. To perform motion compensation of the texture, the coded motion field (MF) needs to be *inverted*.

for highly scalable video coding, many of the properties of the proposed scheme are also beneficial for single-layer coding. Preliminary versions of the proposed method have been presented in our earlier work [22], [23]. In this paper, however, these initial ideas are refined and augmented with the following additional contributions:

- We propose a novel *hierarchical spatio-temporal breakpoint inducing scheme* (HST-BPI), which allows to temporally induce breakpoints from coarser to finer temporal scales – a process which enables further quantization of motion discontinuity information (Sect. IV);
- We derive an analytical model which gives insight into how the importance of texture and motion data changes across temporal scales of the proposed spatio-temporal transform (Sect. VII);
- More thorough experimental validation is presented, with more video sequences at different resolution levels. We further provide initial R-D comparisons with SHVC.

II. Overview

The proposed bidirectional hierarchical anchoring of motion fields represents a fundamental change in the way the temporal transform is performed. We start with a brief overview over the proposed temporal transform, and summarize some of its major benefits; we use Fig. 1 to guide the overview. The aim is to predict frame f_{2k+1} from the neighbouring frames f_{2k} and f_{2k+2}.

The name "hierarchical anchoring" stems from the fact that motion fields are anchored at the reference frames, as opposed to the traditional anchoring in the target frames. As we shall see, this anchoring allows to *predict* all but the coarsest temporal level motion field of a group of pictures from other motion fields, which results in a significant saving when it comes to coding motion fields.

The *key* enabling feature of the proposed temporal transform is that *discontinuities* in the motion field travel with foreground objects. We describe a way of representing motion discontinuities and how to transfer them across spatio-temporal levels in Sect. IV. In order to be useful for predicting target frames, relevant motion fields need to be inverted. As we will see

in Sect. V, this inversion process reveals important properties the motion is undergoing. In particular, information about disoccluded regions, are readily observed. We show how this useful information can be used to guide the lifting predict and update steps of the temporal transform in Sect. VI; note that these prediction modes are communicated as side information in conventional schemes.

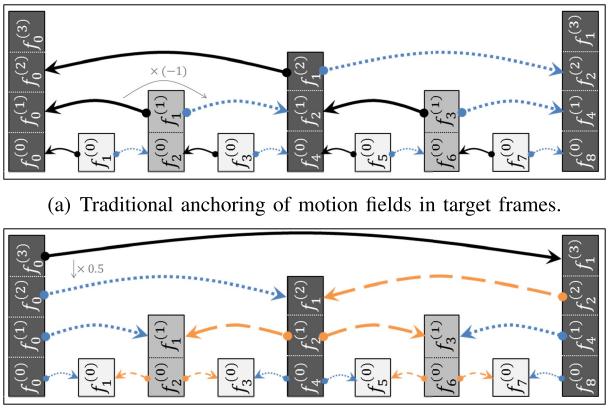
The utility of the proposed scheme increases with the number of temporal decompositions. The temporal subbands of texture and motion are then subjected to a 2D spatial wavelet transform, and coded using JPEG2000. Sect. VII presents how to scalably allocate the rate for all the different spatio-temporal subbands of texture, motion, and breakpoints.

III. Bidirectional Hierarchical Anchoring of Motion Fields

Throughout this paper, we assume the use of a 5/3 temporal wavelet decomposition, based on motion compensated lifting steps. At any given temporal level in this transform, the odd indexed frames are predicted using the preceding and proceeding even indexed frames, while even indexed frames are updated using the prediction residuals from their even indexed temporal neighbours. The even indexed frames are interpreted as a low-pass temporal subband and the procedure is recursively applied to this subband for a total T levels. This transform is common in the literature [15].

We write $M_{i\to j}$ for a motion field that is *anchored* at frame i, *pointing* to frame j; by this we mean that each location $\boldsymbol{x}$ in frame i has an associated motion vector $\boldsymbol{m}_{i,j,\boldsymbol{x}}$ which points to a (corresponding) location $\boldsymbol{x}-\boldsymbol{m}_{i,j,\boldsymbol{x}}$ in frame j. That is $f_i(\boldsymbol{x}) \approx f_j(\boldsymbol{x}-\boldsymbol{m}_{i,j,\boldsymbol{x}})$ for each $\boldsymbol{x}$. Current state-of-the-art codecs (e.g., H.264, HEVC, including their scalable extensions), anchor motion fields at the (odd indexed) target frames of the temporal transform's prediction step; we refer to this as the *traditional anchoring* scheme. In this work, we *hierarchically* anchor motion fields at the (even indexed) reference frames, which we refer to as *bidirectional hierarchical anchoring* (BIHA). As we shall see later, this has some major advantages over the traditional motion field anchoring scheme. Note that the BIHA scheme requires inversion of motion fields so that they can be used for temporal prediction of the target frames. Fig. 2 shows the traditional and the proposed *BIHA motion field* schemes for $T=3$ temporal transform levels.

We use the terms *scaled* and *inferred* to refer to motion fields that serve as prediction references for motion field coding within the temporal hierarchy. In the proposed approach, the motion vectors of each motion field found at level t in the hierarchy are scaled by $\frac{1}{2}$ to form prediction references for a motion field at the next finer level $t+1$. Scaled prediction references are also commonly used in the traditional motion anchoring approach, where the motion vectors of each backward pointing motion field are scaled by -1 to serve as a prediction for the forward pointing motion field anchored at the same frame. The scaled motion fields are shown as blue dotted lines in Fig. 2. As prediction references, these scaled motion fields can be expected to be most efficient under constant (non-accelerated) motion.



(a) Traditional anchoring of motion fields in target frames.

(b) Proposed anchoring of motion fields at the reference frames.

Fig. 2. Two ways of anchoring motion fields: (a) Traditional anchoring at target frames; (b) The proposed *bidirectional hierarchical anchoring* at reference frames. Solid black arrows are *full* motion fields, dotted blue are *scaled* motion fields, and dashed orange indicates *inferred* motion fields.

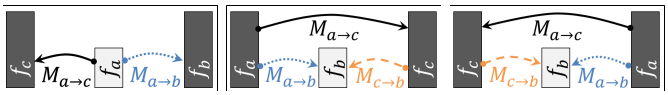


Fig. 3. Frame naming conventions. The target frame (in the middle) is predicted from its temporal left and right neighbour. (a) Trad. anchoring. (b) Prop. anchoring 1. (c) Prop. anchoring 2.

In the proposed scheme, roughly *half* of all motion fields are *inferred* (dashed orange arrows in Fig. 2b); they are specific to our proposed hierarchical motion anchoring scheme, being obtained through composition and inversion of other motion fields at the same and coarser levels of the hierarchy (see Sect. III-A). Importantly, the inferred motion fields can be highly effective in predicting actual motion, even under accelerating conditions.

To facilitate the discussion, we label the frames and motion fields involved in the bi-directional prediction process at any given temporal level t as shown in Fig. 3. Both the traditional and proposed schemes involve a *full* motion field that is either independently coded, or differentially coded at a coarser temporal level. Note that in the BIHA scheme, there are two different arrangements of frames (Fig. 3b/c), depending on the index of the target frame.

A. Motion Field Inference

In the proposed scheme, the fact that the motion fields are anchored at reference frames allows us to notionally *infer* $M_{c\to b}$ from $M_{a\to c}$ and $M_{a\to b}$ as

$$\hat{M}_{c\to b} = M_{a\to b} \circ (M_{a\to c})^{-1}. \tag{1}$$

Note that in the presence of different moving objects, none of the motion fields are truly invertible because of disocclusions at moving object boundaries. However, we propose a well-defined breakpoint dependent procedure for inferring these motion fields in Sect. V-B.

B. Potential for Motion Field Inference in the Traditional Anchoring Scheme

As a prediction tool, the inferred motion fields we compose in the proposed scheme are very appealing since they are very sparse – as we will see in Sect. V, they can be expected to be non-zero only in disoccluded regions. In this paper, motion field inferring is developed entirely for the proposed scheme. It is reasonable to ask whether the traditional scheme could potentially benefit from a similar approach.

As we shall see, the proposed approach makes it possible not only to infer motion fields, but also to deduce regions of disocclusion, where information in the target frame is not observable in one of the source frames. Without some form of explicit encoding, it is not clear how such information can be deduced in the traditional approach. It is important to note that it is not possible to have both scaled and inferred motion fields in the traditional scheme. Motion field scaling is a simple and effective mechanism to generate a prediction reference from another motion field that is anchored at the *same* frame. In the traditional approach, motion fields are anchored at the target frames, so that with the terminology of Fig. 3a, scaling can only be used to predict $M_{a\to b}$ from $M_{a\to c}$ or vice-versa.

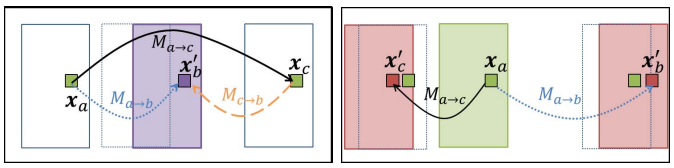
Motion field inference could be used with the traditional anchoring scheme. In particular, with the terminology of Fig. 3a, $M_{a\to b}$ could be inferred from $M_{a\to c}$ and a coarser level motion field, or $M_{a\to c}$ could be inferred from $M_{a\to b}$ and a coarser level motion field, both of which are *alternatives* to motion scaling, but not complementary. Of course, to do this, the coarser level motion fields would also need to be encoded. The proposed approach has the benefit that scaling provides a robust motion prediction mechanism from coarse to fine levels, which is complemented by inference of the remaining finer level motion information, so that motion information need only be coded directly at the very coarsest level of the temporal hierarchy. Regardless of employing motion scaling or inference, the traditional anchoring requires one fully coded motion field (Fig. 2a) at each temporal level; in contrast, there is only *one* fully coded motion field in the BIHA scheme (Fig. 2b).

C. Differential Coding of Motion Fields

The *scaled* and *inferred* motion fields serve as references $\hat{M}_{j\to i}$ for predictive coding of the actual motion field $M_{j\to i}$

$$\Delta_{M_{j\to i}} = M_{j\to i} - \hat{M}_{j\to i}. \tag{2}$$

Clearly, the quality of these *scaled* and *inferred* prediction references has a large impact on the motion coding cost. For scaled motion fields, the *scaled motion residual* $\Delta_{M_{j\to i}}$ represents the *acceleration* between the three frames involved; *inferred motion residuals*, however, are expected to be non-zero only in regions that get disoccluded between frames f_a and f_c. The more temporal levels there are, the more efficient the scheme becomes, since the scaled and inferred residuals can be expected to become smaller at finer temporal levels. We use the term "highly" scalable to highlight the fact that the number of scalability levels do not need to be decided upon at the encoder.



(a) (b)

Fig. 4. Illustration of how quantization of motion fields affects the motion-compensated prediction process. In the proposed BIHA scheme, the *inferred* motion field $M_{c\to b}$ "follows" whatever error there is in the *scaled* motion field $M_{a\to b}$. In contrast, $M_{a\to b}$ and $M_{a\to c}$ are not linked in traditional anchoring, which leads to ghosting if the motion is quantized. (a) BIHA. (b) Traditional anchoring.

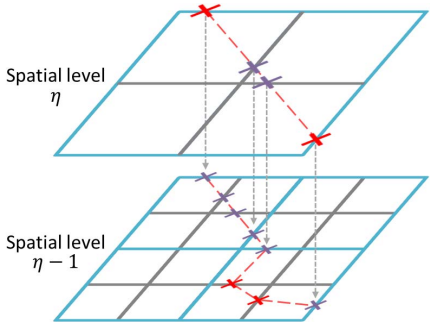


Fig. 5. Scalable geometry representation: Two breakpoints on the *perimeter* of the same *cell* can induce discontinuity information onto the *root* arcs (purple crosses). If the root arc contains a vertex (red cross), the inducing is stopped.

D. Geometrical Consistency of Motion Fields

As bits are discarded from a scalable bit-stream, small prediction residuals in $\Delta_{M_{j\to i}}$ will be quantized to zero so that the motion obtained by the scaling and inference algorithms comes to dominate the visual properties of the reconstructed video. The proposed anchoring of motion fields at reference frames might appear counter-intuitive, because all motion fields have to be transferred to the target frames before motion-compensated prediction can be performed. One key insight of the proposed scheme is that because the *inferred* motion fields "follow" their *scaled* temporal sibling, the warped (inverse) motion fields which are anchored at the target frame and used for the prediction of the target frame f_b lead to geometrically consistent predictions, as shown in Fig. 4.

By contrast, in the case of the traditional anchoring, nothing guarantees that the forward and backward pointing motion fields point to the same geometrical location in the reference frames once the motion gets quantized.

IV. Hierarchical, Spatio-Temporal Induction of Discontinuity Information

The proposed spatio-temporal transform uses discontinuities in the motion field to reason about scene geometry during different stages of the motion field inversion process (see Sect. V). We use so-called *breakpoints* [21] as a highly scalable representation of motion discontinuity information in both resolution and precision. In this work, we extend the scalability attributes of breakpoints to the temporal domain,

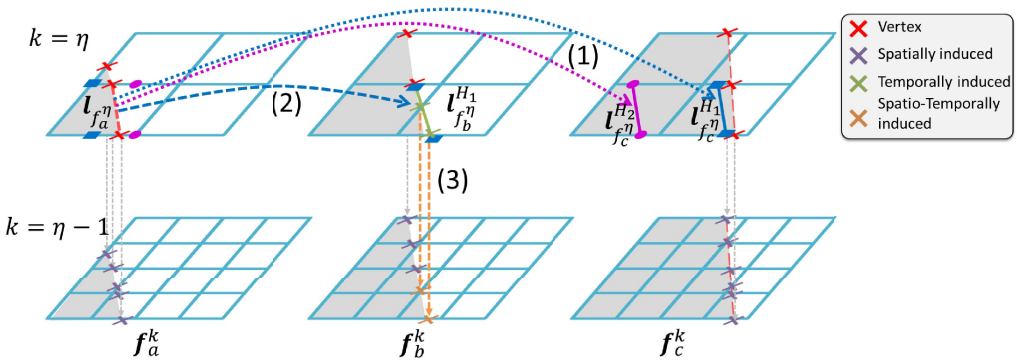


Fig. 6. Spatio-temporal induction of breakpoints: at any given spatial resolution η, the proposed temporal induction process consists of three steps: (1) Assessment of temporal compatibility of line segments induced by breakpoints between two coarse-level frames f_a^η and f_c^η; (2) Warping of *compatible* line segments to f_b; (3) Spatial induction of all breakpoints to the next finer spatial resolution $\eta - 1$. For better visualization, root arcs are not shown in this figure.

which we refer to as *hierarchical spatio-temporal breakpoint induction* (HST-BPI). We start by reviewing how spatial breakpoint induction works, introducing useful terminology, and then present the proposed temporal extension.

A. Spatial Induction of Breakpoints

Breakpoints are organized in a hierarchical manner, such that breakpoints at finer spatial levels can be *induced* from coarser levels. The technical details on how breakpoints are estimated in an R-D optimized way can be found in [21]. Here, we summarise how geometry information can be induced from an existing breakpoint field. We use Fig. 5 to aid the description.

The breakpoint field at spatial level η consists of squares of size $2^\eta \times 2^\eta$ called *cells*, which are the fundamental unit used for inducing discontinuities. A cell consists of four *perimeter arcs* (cyan lines in Fig. 5), as well as two *root arcs* (grey lines in Fig. 5). The significance of root arcs is that they do not exist at coarser levels in the pyramid. Each arc can contain at most one breakpoint. If a cell contains exactly two perimeter breakpoints, and the root arcs at this level have no explicitly coded breaks, connecting the two perimeter breaks allows breakpoints to be induced onto the root arcs. To avoid confusion, we use the term *vertices* to identify the explicitly coded breaks. What this means then is that spatial induction transfers discontinuity information recursively from coarser level vertices to finer levels in the hierarchy, except where such transfer would be in conflict with finer level vertices.

Since each arc in the hierarchy may have a coded vertex, the breakpoint representation is described by a vertex field that is scalable in precision. At lower bitrates, the representation is necessarily highly sparse, with most breaks being induced. Even so, however, signalling a sparse set of vertices at low precision can still occupy a significant portion of the bitrate budget in some video compression applications. In the following, we extend the existing spatial breakpoint induction to a spatio-temporal breakpoint induction, which allows breakpoint fields at finer temporal scales to be completed/improved with breakpoint information from coarser temporal levels.

B. Interaction Between Spatial and Temporal Breakpoints

A natural outcome of the proposed hierarchical coding framework is that at the decoder, the precision of texture, motion, and breakpoint data is higher at coarser temporal levels t (see Sect. VII); the same is true for spatial resolutions η. One can therefore expect that at lower bit-rates, few if any *vertices* will appear at the finer spatio-temporal resolution levels. Since both spatial and temporal induction processes are of interest, we must be able to resolve conflicts between the induced breakpoints that may arise. In this work, we perform induction in a particular sequence, in which breakpoints are induced to all spatial levels of the frames at a particular level in the temporal hierarchy, before moving to the next finer temporal level.

By definition, breakpoints lie on discontinuities in the motion field. One key issue in temporally inducing breakpoints is to find the foreground motion, as this is the motion that we can expect motion boundaries to follow. The foreground motion can be identified through some elementary breakpoint compatibility tests, after which the discontinuity information can be mapped to the target frame. The main question that arises here is what to do if there are already breakpoints present in the target frame. In the following, we explain the main steps of the proposed hierarchical spatio-temporal breakpoint induction (HST-BPI) process, which addresses the aforementioned considerations.

1) Breakpoint Compatibility Check: The first step is to use the two coarser temporal level reference frames f_a and f_c to find *temporally consistent* line segments. For the following discussion, we write f_j^η to denote a spatial level η for frame f_j. We explain the proposed breakpoint warping procedure with the aid of Fig. 6, where a grey object moves from left to right on a static background (white). The procedure to assess the compatibility of breakpoints is motivated by the observation that motion discontinuities travel with the foreground object. Traversing all cells in f_a^η, we identify line segments wherever there are two breaks belonging to the same cell. Each such line segment $\boldsymbol{l}_{f_a^\eta}$ is warped to f_c^η under the hypotheses H_k ($k \in \{1, 2\}$) that the motion on side k of the breakpoint is the foreground motion. The line segment $\boldsymbol{l}_{f_a^\eta}$

is marked *compatible* under hypothesis H_k if each endpoint of the warped line segment $\boldsymbol{l}_{f_c^\eta}^{H_k}$ lies within one arc length of breakpoint in f_c^η. The line segment is considered to be *temporally consistent* if it is compatible with exactly one of the two hypotheses.

2) Warping of Temporally Consistent Line Segments: Once all consistent line segments at a given spatial level η have been discovered, they are warped to the target frame f_b using the relevant compatible motion hypothesis. Note that the motion used to establish compatibility is that between f_a^η and f_c^η, identified as $M_{a\to c}^\eta$, while that used for warping is the separate motion field $M_{a\to b}^\eta$, both of which are drawn from the underlying scalable motion representation at the resolution in question. For each intersection of the warped line segment with an arc in f_b, we check whether the temporally induced break falls into a *spatial break occupied* (SBO) cell. A cell is considered SBO if it contains at least one *spatial break*; by this we mean a break that arises directly or by spatial induction from vertices found only in the same frame f_b. If the cell is empty or only contains temporally induced breaks, the temporal break is registered at the position of the intersection of the line segment with the arc, replacing any existing breakpoint on that arc. Note that spatial breaks can never be replaced by this scheme, because any arc containing a spatial break necessarily belongs to an SBO cell. The breakpoints that are placed on arcs via this line segment warping procedure are termed *temporal breaks*.

The last step of the proposed HST-BPI method applies spatial induction, as explained in Sect. IV-A.

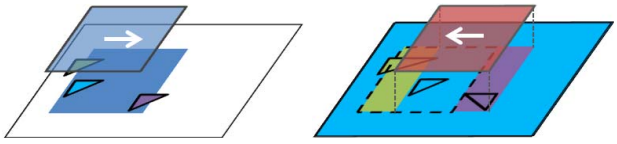
V. Motion Field Operations

In the proposed motion field anchoring scheme, motion fields are used to predict texture information during motion-compensated temporal prediction; they are also used to predict other motion fields at finer levels of the temporal hierarchy. Whenever we use a motion field to predict texture information, this requires us to *invert* the motion field so as to anchor it at the target frame, as shown in Fig. 1. This inversion process is explained in Sect. V-A.

As already noted, a key feature of the proposed approach is that motion fields at coarse levels can be used to recursively predict the motion found at finer levels. Half the motion fields at a given level in the temporal hierarchy are predicted simply by scaling a coarser level motion field anchored at the same frame. The other half of the motion fields are predicted using the more complex operation that we have called *motion inference*. As revealed by Eq. 1, motion inference involves composition and inversion. However, we find it more useful to consider motion inference as a *unified operation*; this is the subject of Sect. V-B.

A. Inversion of Motion Fields

We generate $M_{j\to i}$ from the available motion field $M_{i\to j}$ using the *cellular affine warping* (CAW) procedure presented in [23]. In brief, f_i is partitioned into small triangles and the motion $M_{i\to j}$ associated with each such triangle is warped to the *target* frame f_j using an affine flow model. In order to guarantee that the warped triangles cover the support of f_j, we extend the triangular mesh and associated motion field in f_i by one pixel in each direction, assigning zero motion to the extended locations.



(a) (b)

Fig. 7. The proposed CAW method for inverting motion fields readily observes disocclusion (green) and folding (magenta) in the target frame f_j; the obtained Disocclusion and Folding (DF) mask is valuable to guide the bidirectional prediction process. (a) $M_{i\to j}$. (b) $\hat{M}_{j\to i}$ + DF mask.

What this means is that each location in f_j will be assigned motion that can be used to predict f_j from f_i; however, some locations might be assigned multiple motion candidates due to *folding* of the triangular mesh. Disoccluded regions are assigned a smooth (stretched) motion field during the inversion process. Each triangle can readily be assigned one of three categories, as illustrated in Fig. 7: visible in both frames (cyan); disoccluded in target frame (green); and folded (double mapping) in target frame (magenta). Fig 7b shows an example disocclusion and folding mask; we remind the reader that this mask is obtained during the motion inversion process, and *no side information* has to be communicated.

Folded regions are identified immediately wherever the cellular warping procedure assigns multiple motion vectors to the same location in f_j. We explain how we resolve such double mappings with the aid of breakpoints in Sect. V-C.

Regions of likely disocclusion can be identified by stretching of triangles from f_i as they are mapped to f_j, where the determinant of the affine transform exceeds a small threshold. We show in Sect. VI how the disocclusion information provides a valuable mechanism to guide the temporal predict and update steps. In particular, in the proposed bidirectional prediction setting, disoccluded regions are predicted from the other reference frame. For the case where a location is occluded in both reference frames, the affine motion has the effect of stretching the texture information, which creates a smooth prediction.

B. Inference of Motion Fields

We remind the reader that the *inference* of motion field $M_{b\to c}$ involves the composition of two motion fields: its temporal sibling $M_{a\to b}$, and its parent motion field $M_{a\to c}$ (naming from Fig. 3). Triangles are formed in frame f_a and warped to frame f_c. We use the CAW procedure explained above to warp the parent motion field from frame f_a to frame f_c; each location $\boldsymbol{x}_c$ is then assigned the difference between the two motion fields, i.e. $\hat{M}_{c\to b}(\boldsymbol{x}_c) = -(M_{a\to c}(\boldsymbol{x}_a) - M_{a\to b}(\boldsymbol{x}_a))$.

By and large, this CAW procedure for motion field inference provides an excellent prediction for the original $M_{b\to c}$ field. In particular, the procedure correctly accounts for acceleration. However, in regions of f_c which correspond to background

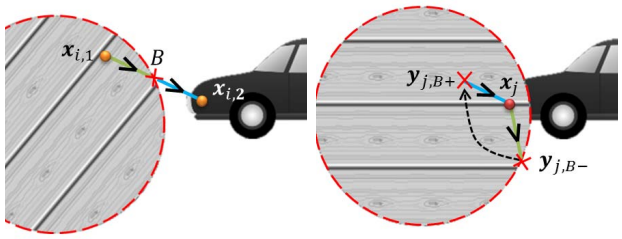


Fig. 8. A disk moves on top of a stationary car. Two points in the reference frame ($x_{i,1}$ and $x_{i,2}$) map to the same point x_j in the target frame. Breakpoints are used to identify the foreground moving object. (a) Reference frame f_i. (b) Target frame f_j.

information that was occluded in f_a, the CAW procedure produces a poor prediction. In frame f_c, these *disoccluded* regions correspond to stretched triangles, where the CAW procedure infers a smooth transition between the background and foreground motions. For these regions, we propose to use a more realistic prediction based on *piece-wise constant motion extrapolation*. We present the essence of this method, and refer the interested reader to [23] for a more detailed explanation.

Whenever a triangle is stretched due to disocclusion (green regions in Fig. 7b), we expect that the stretched triangle intersects with a motion discontinuity in the target frame, which partitions the triangle into another triangle and a quadrilateral. While the term "background" might not have absolute semantic associations, its significance is that the background region should always correspond to the part of the triangle which has been stretched, while the other portion should be very small. Using the fact that motion discontinuities travel with the foreground object, we identify which part of the disoccluded triangle belongs to the background, and then extrapolate the identified background motion up to the motion discontinuity. Clearly, this procedure could be improved by analysing the neighbourhood motion and creating a higher order motion extrapolation.

C. Resolving Double Mappings in Folded Regions

As the CAW procedure visits triangles in f_i and maps motion vectors from $M_{i\to j}$ into frame f_j, a location x_j might already have an assigned motion, which is a result of folding in the motion field. That is, there are two locations $x_{i,1}$ and $x_{i,2}$ such that $x_{i,1}+M_{i\to j}(x_{i,1})$ and $x_{i,2}+M_{i\to j}(x_{i,2})$ are both equal to x_j. We disambiguate such double mappings by observing that the motion discontinuity "moves" with the foreground object. Fig. 8 illustrates the proposed procedure, where a disk moves on top of a stationary car.

Consider the line segment that connects $x_{i,1}$ with $x_{i,2}$ in f_i. For the majority of cases, this line intersects with (at least) one motion discontinuity, which we denote as B.[1] Let $y_{i,s} = (1-s)x_{i,1}+sx_{i,2}$ be a parametrisation of the points on this line segment, where $s \in [0,1]$, and let us consider the behaviour of $y_{j,s} = y_{i,s} + M_{i\to j}(y_{i,s})$ as s transitions from 0 to 1. When $y_{i,s}$ arrives at the break location $y_{i,B}$, the mapped location $y_{j,s}$ will exhibit a discontinuous jump (black dotted arrow in Fig. 8b). Let $y_{i,B-}$ and $y_{i,B+}$ denote the locations immediately before and after the break, having mapped locations $y_{j,B-}$ and $y_{j,B+}$. We expect one of these two mapped locations to align (at least very closely) with a break location in the target frame f_j. Again, this is because motion discontinuities travel with the foreground object. Accordingly, we conclude that $M_{j\to i}(x_j) = x_{i,1} - x_j$ if $y_{j,B-}$ lies closer to a motion discontinuity in the target frame than $y_{j,B+}$, else $M_{j\to i}(x_j) = x_{i,2} - x_j$.

In this paper, we consider breakpoints that lie within one arc length of $y_{j,B-}$ and $y_{j,B+}$. Where both or neither of $y_{j,B-}$ and $y_{j,B+}$ appear to coincide with breaks in the target frame, we currently record them as unresolved; such locations are assigned an average value from all adjacent neighbours with valid assignments.

[1] If there are multiple motion discontinuities between $x_{i,1}$ and $x_{i,2}$, we record the closest intersection with a discontinuity boundary for both $x_{i,1}$ and $x_{i,2}$.

VI. Motion-Compensated Temporal Filtering

Both original and inverted motion fields are used together with *discovered* information about disoccluded regions, to drive a motion compensated temporal lifting transform of the video texture information. The temporal transform employed in this work is composed of two parts: 1) bidirectional *prediction* of the target frame from its temporal neighbours; and 2) a temporal *update* step, which feeds some of the motion compensated residual from the prediction step back to the reference frames. This update step helps reduce temporal aliasing in case finer temporal levels are discarded from the bit-stream and also has fundamental benefits in reducing the impact of quantization noise in the temporal subbands on reconstructed video quality [24].

In the proposed framework, the prediction step uses inverted motion fields ($\hat{M}_{b\to a}$ and $\hat{M}_{b\to c}$ from Fig. 2), while the update step uses the original motion fields. This differs markedly from traditional approaches, where bi-directional prediction is performed using original (encoded) motion fields. However, the proposed approach has the advantage that disocclusion information, *discovered* during the inversion process, can inform the prediction process. For the following discussion, we define a *disocclusion* mask $S_{j\to i}$ as follows:

$$S_{j\to i}(x) = \begin{cases} 0 & x \text{ disoccluded in } (M_{i\to j})^{-1} \\ 1 & \text{otherwise.} \end{cases} \tag{3}$$

In a bidirectional prediction setup, we compute two such disocclusion masks: one during the inversion of $M_{a\to b}$, and the other one while inverting $M_{c\to b}$. These masks are denoted $S_{b\to a}$ and $S_{b\to c}$, respectively.

A. Prediction Step

Let $\mathcal{W}_{\hat{M}_{i\to j}}(f_j)$ denote the warping process of frame f_j to frame f_i. That is, $f_{j\to i}(x) = \left(\mathcal{W}_{\hat{M}_{i\to j}}(f_j)\right)(x)$ at each location x. At each x in frame f_b, the prediction $\hat{f}_b(x)$ is formed using $\hat{M}_{b\to a}$ and $\hat{M}_{b\to c}$, together with the *estimated*

disocclusion masks $S_{b\to a}$ and $S_{b\to c}$, as:

$$\hat{f}_b(\boldsymbol{x}) = \begin{cases} \dfrac{S_{b\to a}(\boldsymbol{x}) f_{a\to b}(\boldsymbol{x}) + S_{b\to c}(\boldsymbol{x}) f_{c\to b}(\boldsymbol{x})}{\kappa(\boldsymbol{x})} & \kappa(\boldsymbol{x}) > 0 \\ 0.5\big(f_{a\to b}(\boldsymbol{x}) + f_{c\to b}(\boldsymbol{x})\big) & \kappa(\boldsymbol{x}) = 0, \end{cases} \tag{4}$$

where $\kappa(\boldsymbol{x}) = S_{b\to a}(\boldsymbol{x}) + S_{b\to c}(\boldsymbol{x})$. The prediction residual Δf_b then becomes $\Delta f_b = f_b - \hat{f}_b$, which is also the high-pass temporal subband.

B. Update Step

Motion fields are invertible everywhere except at disoccluded regions. We use the disocclusion masks $S_{b\to j}$, where $j \in \{a, c\}$ are the previous and future reference frames, to disable the update step in disoccluded regions. The updated frame becomes

$$f_j^{\text{updated}}(\boldsymbol{x}) = f_j(\boldsymbol{x}) + \beta S_{b\to j}(\boldsymbol{x}) \Delta f_{b_{b\to j}}(\boldsymbol{x}), \tag{5}$$

where the update weight β is defined as:

$$\beta = \begin{cases} 0.25 & \kappa(\boldsymbol{x}) = 2 \\ 0.5 & \kappa(\boldsymbol{x}) < 2. \end{cases} \tag{6}$$

In other words, a quarter of the prediction residual is fed back to the two reference frames in regions that are visible from both references. If a location is only visible from one side, the temporal transform is effectively reduced from the bi-orthogonal 5/3 transform to a 2-tap Haar transform. If a region is disoccluded in both frames, Eq. 5 eliminates the update step; in such regions the prediction step in Eq. 4 averages two spatially smooth predictors.

VII. Scalable Rate Allocation

In order to quantitatively evaluate the proposed anchoring and compare it to the traditional anchoring of motion fields, we have to code the texture, motion, and breakpoint data. For this evaluation to be meaningful, we need to balance the error contributions of the different data types, which requires an understanding on how errors propagate. In the following, we present an analytical model that allows us to understand how quantization errors in the motion field subbands impact distortion in the final reconstructed video sequence. We then consider the impact of errors in the texture and breakpoint information in Sect. VII-B.

A. Motion Error

We first investigate for the different types of motion fields how a quantization error of $\boldsymbol{\delta}$ propagates across frames. Let $\boldsymbol{f}^{(t)}$ denote the frames produced after $T - t$ levels of temporal synthesis. For any level $t \in [0, T-1]$ of the temporal transform, motion fields $\boldsymbol{m}^{(t)}_{i,j,\boldsymbol{x}}$ are used to synthesize frames $\boldsymbol{f}^{(t)}$ from frames $\boldsymbol{f}^{(t+1)}$ together with high-pass temporal subband frames $\boldsymbol{d}^{(t+1)}$.

For any level t, the reconstructed frames at the finest temporal level can always be expressed as

$$\boldsymbol{f}^{(0)} = S_L^{(t+1)}(\boldsymbol{f}^{(t+1)}) + \sum_{p=1}^{t+1} S_H^{(p)}(\boldsymbol{d}^{(p)}), \tag{7}$$

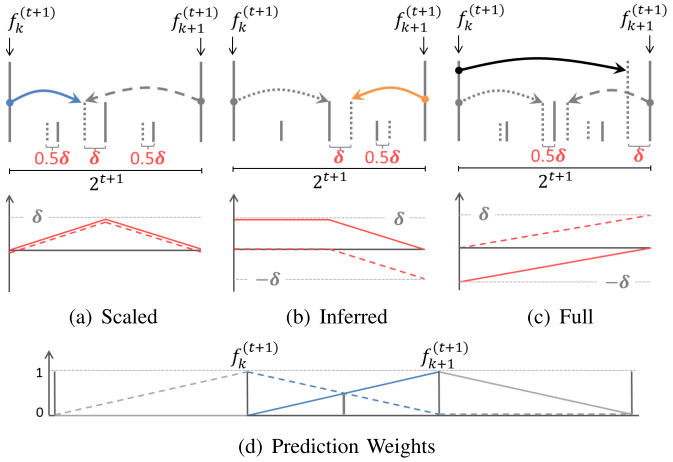


Fig. 9. Illustration how errors in motion fields spread. The red solid line shows how the texture data from the right reference frame is affected by introducing an error $\boldsymbol{\delta}$ into a motion field; the red dashed line shows the same for the left reference frame. (d) shows the prediction weight of the two reference frames.

where $S_L^p(\cdot)$ and $S_H^p(\cdot)$ denote low- and high-pass temporal synthesis operators associated with the information injected at level p in the transform.

For convenience of analysis, consider for the moment that there is a constant displacement error $\boldsymbol{\delta}$ in $\boldsymbol{m}^{(t)}_{i,j,\boldsymbol{x}}$. Our goal is to understand the impact of this error on the total squared error distortion in the reconstructed video. This reconstructed video distortion arises from geometric distortions (i.e., spatial shifts) in each of the synthesized texture contributions found in Eq. 7. In particular, the synthesis operators $S_L^{(p)}$ and $S_H^{(p)}$, $p \le t+1$ each depend on $\boldsymbol{m}^{(t)}_{i,j,\boldsymbol{x}}$ and hence on the error $\boldsymbol{\delta}$, so that the total error experienced from all frames at the finest temporal level becomes

$$\Delta \boldsymbol{f}^{(0)} = D_L^{(t+1)}(\boldsymbol{f}^{(t+1)}, \boldsymbol{\delta}) + \sum_{p=1}^{t+1} D_H^{(p)}(\boldsymbol{d}^{(p)}, \boldsymbol{\delta}). \tag{8}$$

While it is possible to analyze each of these contributions separately, it turns out that the major contribution to $\Delta \boldsymbol{f}^{(0)}$ arises from the first term $D_L^{(t+1)}(\boldsymbol{f}^{(t+1)}, \boldsymbol{\delta})$, corresponding to distortion arising for the low-pass synthesis operation. In the following analysis, we therefore assume that all detail bands $\boldsymbol{d}^{(p)}$ are zero. In order to simplify the ensuing analysis, we consider only the case where the original motion is 0, so that the error $\boldsymbol{\delta}$ becomes the distorted motion field. The resulting analysis remains valid for any translational original motion field, and is an excellent approximation for more general original motion fields.

Fig. 9 shows the effect of an error $\boldsymbol{\delta}$ in any level t scaled, inferred or full motion fields found between frames $f_k^{(t+1)}$ and $f_{k+1}^{(t+1)}$. The second row in the figure shows how the texture contributions from these left (dashed line) and right (solid line) reference frames become shifted by the time they reach the finest temporal level $\boldsymbol{f}^{(0)}$. Fig. 9d shows the relative contribution from each of $f_k^{(t+1)}$ and $f_{k+1}^{(t+1)}$ to each frame of the final synthesized video sequence.

In the following, we provide an asymptotic analysis of the impact of $\boldsymbol{\delta}$ in the limit as t becomes very large so that

the synthesized video can be treated as continuous in time. As revealed by Fig. 9, the motion errors in question result in shifted contributions that can produce reconstructed video errors only at frame times $(k+\tau)2^{t+1}$, where $\tau \in [0,1]$. We can express these error frames in terms of their contributions from the left and right reference frames as $\Delta f_\tau^{(0)} = \Delta f_{\text{left},\tau}^{(0)} + \Delta f_{\text{right},\tau}^{(0)}$.

1) Scaled Motion Fields: The scaled motion field, shown as a solid blue arrow in Fig. 9a, has motion vectors $\boldsymbol{m}_{2k,2k+1,\boldsymbol{x}}^{(t)}$. Introducing an error of $\boldsymbol{\delta}$ to these motion vectors yields an error contribution $\Delta f_{\text{left},\tau}$ that can be expressed in the Fourier domain as

$$\Delta \hat{f}_{\text{left},\tau}(\boldsymbol{\omega}) = \hat{f}(\boldsymbol{\omega})\big(1 - e^{-j\boldsymbol{\omega}^t\boldsymbol{\delta}2\tau}\big)(1-\tau), \tag{9}$$

over the interval $\tau \in [0, 0.5]$. Here, $\hat{f}(\boldsymbol{\omega})$ is the Fourier transform of $f_k^{(t+1)}$, which is identical to $f_{k+1}^{(t+1)}$ under our zero original motion assumption. As shown in Fig. 9a, the same shifts arise in the contribution from $f_{k+1}^{(t+1)}$ due to the motion inference process. Accordingly,

$$\Delta \hat{f}_{\text{right},\tau}(\boldsymbol{\omega}) = \hat{f}(\boldsymbol{\omega})\big(1 - e^{-j\boldsymbol{\omega}^t\boldsymbol{\delta}2\tau}\big)\tau. \tag{10}$$

Thus, for $\tau \in [0, 0.5]$, the total error at frame $f_\tau^{(0)}$ can be expressed as

$$\Delta \hat{f}_\tau(\boldsymbol{\omega}) = \hat{f}(\boldsymbol{\omega})\big(1 - e^{-j\boldsymbol{\omega}^t\boldsymbol{\delta}2\tau}\big) \approx \hat{f}(\boldsymbol{\omega})2\tau j\boldsymbol{\omega}^t\boldsymbol{\delta}, \tag{11}$$

where we have used a first order Taylor series approximation for the complex exponential.

Evidently, the errors $\Delta \hat{f}_\tau(\boldsymbol{\omega})$ that arise when $\tau \in [0.5, 1]$ are just a mirror image of those that arise $\tau \in [0, 0.5]$. Using Parseval's Theorem, the total energy of the prediction error $|e_{scal}^{\infty}|^2$ can then be expressed as

$$|e_{\text{scal}}^{\infty}|^2 = 2^{t+1}\cdot 2\int_0^{0.5}\frac{1}{(2\pi)^2}\int_{-\pi}^{\pi}\int_{-\pi}^{\pi}|\Delta\hat{f}_\tau(\boldsymbol{\omega})|^2 \mathrm{d}\tau\,\mathrm{d}\boldsymbol{\omega} \tag{12}$$

where the factor of 2^{t+1} arises from the observation that our interval $\tau \in [0,1]$ corresponds to 2^{t+1} reconstructed video frames. This error energy (distortion) can be approximated by

$$\begin{aligned}|e_{\text{scal}}^{\infty}|^2 &\approx 2^{t+4}\int_0^{0.5}\tau^2\mathrm{d}\tau\,|\boldsymbol{\delta}|^2 \\ &\quad\times \underbrace{\frac{1}{(2\pi)^2}\int_{-\pi}^{\pi}\int_{-\pi}^{\pi}|\hat{f}(\boldsymbol{\omega})|^2|\boldsymbol{\omega}|^2\cos^2(\Theta)\mathrm{d}\boldsymbol{\omega}}_{=\frac{1}{2}E[|\nabla f|^2]A\ \text{(assuming isotropic power spect.)}} \\ &= \frac{2^t}{3}E[|\nabla f|^2]A|\boldsymbol{\delta}|^2,\end{aligned} \tag{13}$$

where $E[|\nabla f|^2]$ is the average gradient power, and A is the area of the frame. Let $D_{\boldsymbol{m}^{(t)}}^{\text{scal}}$ denote the total amount of distortion in scaled motion fields at temporal level t. The resulting total amount of distortion in the reconstructed video can then be expressed as

$$D_{\boldsymbol{m}^{(t)}\to \boldsymbol{f}_0}^{\text{scal}} = D_{\boldsymbol{m}^{(t)}}^{\text{scal}}\frac{2^t}{3}E[|\nabla f|^2] = D_{\boldsymbol{m}^{(t)}}^{\text{scal}}\cdot\alpha_{\text{scal}}^{(t)}\cdot E[|\nabla f|^2]. \tag{14}$$

While the model we present assumes a constant error $\boldsymbol{\delta}$, we note that it provides a good approximation also for gradually changing errors. In this paper, we do not specifically consider high frequency motion errors, but note that the sparse motion representation required for good coding efficiency inevitably leads to motion fields (and hence motion quantization errors) that are smooth except in the vicinity of breakpoints.

2) Inferred Motion Fields: The inferred motion field, shown as a solid orange arrow in Fig. 9b, has motion vectors $\boldsymbol{m}_{2k+2,2k+1,\boldsymbol{x}}^{(t)}$. Errors in these motion vectors do not affect the scaled sibling motion field $\boldsymbol{m}_{2k,2k+1,\boldsymbol{x}}^{(t)}$. This can be seen in Fig. 9b, where the dotted red line, indicating the shift in the texture of the left reference frame, is zero over the first half interval $\tau \in [0, 0.5]$. Over this half interval, $\hat{f}_\tau$ can be expressed in the Fourier domain as

$$\Delta\hat{f}_\tau(\boldsymbol{\omega}) = \hat{f}(\boldsymbol{\omega})\big(1-(1-\tau)-\tau e^{-j\boldsymbol{\omega}^t\boldsymbol{\delta}}\big) \approx \hat{f}(\boldsymbol{\omega})\tau j\boldsymbol{\omega}^t\boldsymbol{\delta}. \tag{15}$$

For $\tau \in [0.5, 1]$, the expression is

$$\begin{aligned}\Delta\hat{f}_\tau(\boldsymbol{\omega}) &= \hat{f}(\boldsymbol{\omega})\big(1-(1-\tau)e^{j\boldsymbol{\omega}^t\boldsymbol{\delta}(2\tau-1)} - \tau e^{j\boldsymbol{\omega}^t\boldsymbol{\delta}(2\tau-2)}\big) \\ &\approx \hat{f}(\boldsymbol{\omega})(\tau-1)j\boldsymbol{\omega}^t\boldsymbol{\delta}.\end{aligned} \tag{16}$$

Again, using Parseval's Theorem, the sum of squared errors can be written as

$$\begin{aligned}|e_{\text{inf}}^{\infty}|^2 &\approx \frac{1}{2}E[|\nabla f|^2]A2^{t+1}|\boldsymbol{\delta}|^2\Big(\int_0^{0.5}\tau^2\mathrm{d}\tau + \int_{0.5}^{1}(\tau-1)^2\mathrm{d}\tau\Big) \\ &= \frac{2^t}{12}E[|\nabla f|^2]A|\boldsymbol{\delta}|^2.\end{aligned} \tag{17}$$

Let $D_{\boldsymbol{m}^{(t)}}^{\text{inf}}$ denote the total amount of distortion in inferred motion fields at temporal level t. The resulting total amount of distortion in the reconstructed video can then be expressed as

$$D_{\boldsymbol{m}^{(t)}\to \boldsymbol{f}_0}^{\text{inf}} = D_{\boldsymbol{m}^{(t)}}^{\text{inf}}\frac{2^t}{12}E[|\nabla f|^2] = D_{\boldsymbol{m}^{(t)}}^{\text{inf}}\cdot\alpha_{\text{inf}}^{(t)}\cdot E[|\nabla f|^2]. \tag{18}$$

3) Full Motion Fields: At the coarsest level of the temporal hierarchy, where $t = T-1$, the proposed method involves one full motion field $\boldsymbol{m}_{k,k+1,\boldsymbol{x}}^{(T)}$, anchored at frame f_k^T and pointing to frame f_{k+1}^T, as shown in Fig. 9c. Full motion fields are never used to directly predict their target frame, but all lower level motion fields depend upon them. An error of $\boldsymbol{\delta}$ in a full motion field leads to error contributions

$$\begin{aligned}\Delta\hat{f}_{\text{left},\tau}(\boldsymbol{\omega}) &= (1-\tau)\hat{f}(\boldsymbol{\omega})\big(1-e^{-j\boldsymbol{\omega}^t\boldsymbol{\delta}\tau}\big) \\ \Delta\hat{f}_{\text{right},\tau}(\boldsymbol{\omega}) &= \tau\hat{f}(\boldsymbol{\omega})\big(1-e^{-j\boldsymbol{\omega}^t\boldsymbol{\delta}(\tau-1)}\big)\end{aligned} \tag{19}$$

Each term alone is a stretched version of the corresponding term that we studied in connection with scaled motion fields. Indeed, if we consider only the left or right error contribution in isolation, the total squared error associated with such a contribution turns out to be the same for both full motion field errors and scaled motion field errors at level $t = T-1$. However, the left and right error contributions from an error in the full motion field approximately cancel each other out.

TABLE I

SQUARED ERRORS FOR DIFFERENT TEMPORAL TEXTURE AND MOTION SUBBANDS FOR A TOTAL OF $T = 3$ TEMPORAL DECOMPOSITIONS

t	$\alpha_{\text{text}}^{(t)}$	$\alpha_{\text{scal}}^{(t)}$	$\alpha_{\text{inf}}^{(t)}$	$\alpha_{\text{full}}^{(t)}$
0	0.719	0.5	0.125	–
1	0.922	0.75	0.1875	–
2	1.586	1.375	0.34375	–
3	–	–	–	1.375

This is because geometric shifts in the left contribution are matched by opposing shifts in the right contribution, as seen in the second row of Fig. 9c. It follows that full motion field errors produce significantly smaller levels of reconstructed video distortion than errors in the scaled motion fields. However, the divergent shifts induced in the left and right reference frame contributions to $\boldsymbol{f}^0$ yield substantial levels of "ghosting." By contrast, distortions introduced by errors in scaled motion fields are free from such visually disturbing ghosting artefacts. It would be beneficial to adopt a distortion metric which could specifically account for the objectionable nature of ghosting artefacts; however, the development of such a metric would require subjective evaluations that lie beyond the scope of this paper. As a compromise, therefore, we choose to assign the same weighting factor to both scaled and full motion fields, i.e. $\alpha_{\text{full}}^{(T)} = \alpha_{\text{scal}}^{(T-1)}$, leaving us with the model

$$D^{\text{full}}_{\boldsymbol{m}^{(T)} \rightarrow \boldsymbol{f}_0} = D^{\text{full}}_{\boldsymbol{m}^{(T)}} \cdot \alpha_{\text{full}}^{(T)} \cdot E[|\nabla f|^2]. \tag{20}$$

4) Distortion Scaling Factors for the Discrete Case: The asymptotic analysis above unveils the coupling between motion errors and reconstructed video errors for the proposed motion representation. For small t, where very few video frames lie in the interval $[k2^{t+1}, (k+1)2^{t+1}]$, this continuous analysis is only approximately valid. Actual coupling factors are shown in Table I for $T = 3$ levels of temporal decomposition. Squared quantization errors in individual subbands b of the breakpoint-adaptive spatial wavelet transform that is used to compress motion fields of type "mtyp" are scaled by the overall weighting factor

$$w_{\text{mtyp}}^{(t,b)} = \alpha_{\text{mtyp}}^{(t)} \cdot E[|\nabla f|^2] \cdot G_{\text{mdwt}}^{(b)} \tag{21}$$

in order to discover their impact on reconstructed video distortion; here $G_{\text{mdwt}}^{(b)}$ is the squared Euclidean norm of the spatial DWT synthesis basis functions associated with motion subband b.

B. Rate Allocation With Breakpoint and Texture Errors

Breakpoints and motion are tightly linked, and hence the analysis for errors introduced by quantizing breakpoints is similar to the one presented above. More details can be found in [21]. As we will see in Sect. VIII-B, this approximation leads to a very good performance.

The temporal texture subbands produced by our motion adaptive temporal transform are also subjected to a spatial DWT, whose subband samples are subject to quantization errors. The impact of squared subband quantization errors on distortion in the reconstructed video sequence can be modeled using a separate set of weighting factors

$$w_{\text{text}}^{(t,b)} = \alpha_{\text{text}}^{(t)} G_{\text{tdwt}}^{(b)} \tag{22}$$

where $G_{\text{tdwt}}^{(b)}$ is the squared Euclidean norm of the spatial DWT synthesis basis functions associated with texture subband b and $\alpha_{\text{text}}^{(t)}$ is the squared Euclidean norm of the temporal synthesis basis functions associated with temporal subbands at level t.

Together, these weights are used to drive a rate-distortion optimized rate allocation algorithm. In practice, the rate allocation is performed using the post-compression rate-distortion (PCRD) strategy of JPEG 2000's EBCOT algorithm [25]. That is, each of the individual subbands and breakpoint vertex bands are subjected to embedded block-based coding, collecting distortion-length slopes for each coding pass, after which the block coding passes are arranged into a global set of quality layers based on the distortion-length slopes, weighted using the factors found above. The resulting scalable video bit-stream can be reconstructed at any of the rate-distortion optimal operating points obtained by discarding quality layers from the overall representation.

VIII. EXPERIMENTAL RESULTS AND DISCUSSION

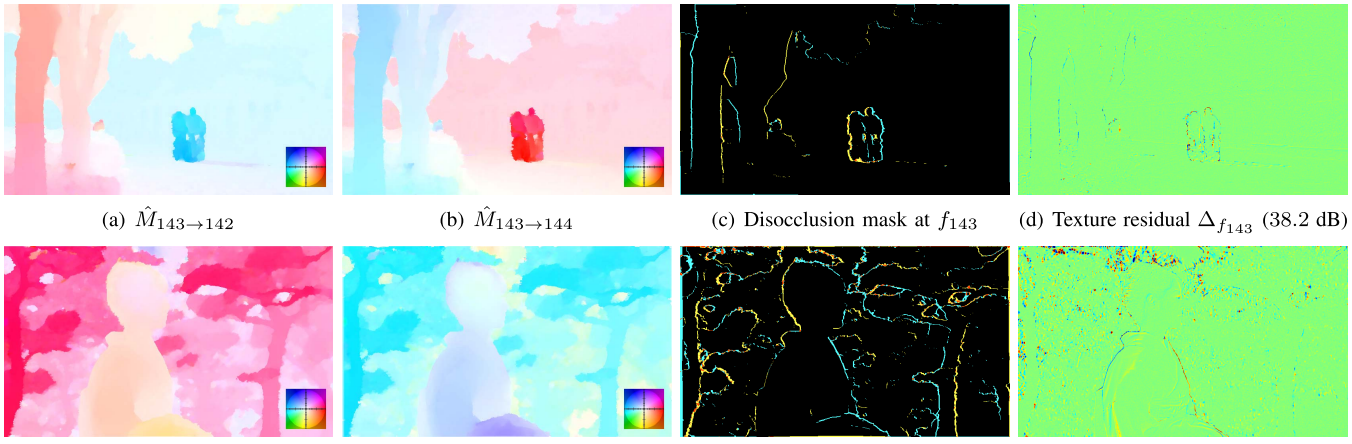
In this section, we evaluate various aspects of the proposed method. In Sect. VIII-A, we show results of the motion field inversion process on common natural test sequences to demonstrate the robustness of the proposed motion inference and temporal prediction framework. These results are particularly encouraging, since they demonstrate the potential of the proposed approach to achieve high quality predictions, using estimated motion anchored at the reference frames. The motion estimates used for this preliminary analysis have the desirable piecewise smooth characteristics that our approach expects.

In Sect. VIII-B, we present results of the novel hierarchical spatio-temporal breakpoint induction method. Sect. VIII-C provides a more comprehensive study of the rate-distortion benefits offered by our proposed approach, including the rate allocation scheme of Sect. VII. This study uses a large collection of synthetic sequences,[2] for which ground truth motion is available.

We have chosen to use ground truth motion for the comprehensive comparison for two reasons. First, it is *instructive* to *decouple* motion *estimation* from the motion *compensation* process, especially since the goal is to compare quite different transform structures, involving motion between different frame pairs. Also, we are not yet in a position to provide reliable experimental results with a complete set of hierarchically structured motion fields that are estimated.[3] Ideally, motion fields employed in this work should exhibit hierarchical consistency, including the property that motion fields anchored at the same frame should share the same set of breakpoints. This property is inherently satisfied by any valid ground truth motion field and could be introduced into motion estimation

[2] Available on: http://ivmp.unsw.edu.au/~dominicr/biha_scheme.html

[3] The estimated motion used successfully in Sect. VIII-A involves only one level of temporal transform.



(a) $\hat{M}_{143\to142}$ (b) $\hat{M}_{143\to144}$ (c) Disocclusion mask at f_{143} (d) Texture residual $\Delta_{f_{143}}$ (38.2 dB)

(e) $\hat{M}_{7\to6}$ (f) $\hat{M}_{7\to8}$ (g) Disocclusion mask at f_7 (h) Texture residual Δ_{f_7} (33.4 dB)

Fig. 10. Qualitative results of the proposed motion field inversion process on two natural sequences (Park and Kimono). The parent motion field $M_{a\to c}$ is estimated using [6]; Columns 1 and 2 show (color-coded) $\hat{M}_{b\to a}$ and $\hat{M}_{b\to c}$, obtained by *inverting* the *scaled* forward ($\hat{M}_{a\to b}$) and *inferred* backward ($\hat{M}_{c\to b}$) pointing motion fields; (c) and (g) show the union of the disocclusion masks computed during the motion field inversion process (black means the texture is visible from both reference frames, yellow and cyan show regions that are occluded in the left and right reference frame, respectively). (d) and (h) show heatmaps (clipped to $[-0.2, 0.2]$ for visualization) of the texture residual after bidirectional prediction, where green means no error, and blue and red indicate negative and positive prediction errors.

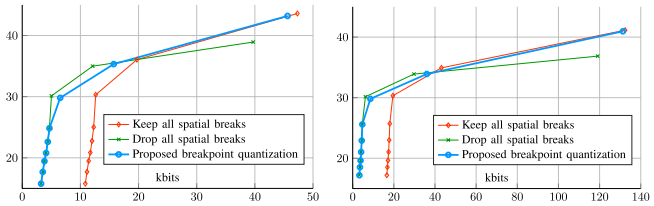
schemes in the future. This is an interesting and parallel stream of research that is beyond the scope of the present paper. To provide some evidence of the applicability of the scheme on real sequences, we apply the method on two natural sequences for which motion estimated using a readily available optical flow estimator [6] almost satisfies the hierarchical consistency.

A. Motion Field Inversion and Texture Prediction

We show results of the proposed motion inversion and bidirectional texture prediction framework, which is a key element of the proposed method. Fig. 10 shows *inverted* motion fields for two common test sequences (e.g., Park and Kimono). The motion $M_{a\to c}$ between the two reference frames is estimated using the motion-detail preserving optical flow estimator from [6], using the default parameters. We highlight that in this experiment, *no residuals* are coded at the finer temporal level; in other words, the *scaled* and the *inferred* motion field residuals, as well as all spatial breakpoints and texture information of the target frame are quantized to zero. This operating mode can be seen as temporal upsampling, as presented in [26]. The quality of the inverted motion fields is evidenced by the generally low prediction residuals; this demonstrates the success of motion inversion and occlusion handling.

B. Hierarchical Spatio-Temporal Breakpoint Induction

This section evaluates the hierarchical spatio-temporal breakpoint induction (HST-BPI). As mentioned in Sect. IV, the aim of HST-BPI is to *improve* existing (spatial) breakpoint fields. At very high bit-rates, high quality spatial breakpoint fields are anchored at the target frames, and temporal induction should ideally not change anything. At medium to low bit-rates, only few or no spatial breakpoints might be decoded at fine temporal levels; in this case, the scheme completely relies on temporally induced breakpoints.



(a) (b)

Fig. 11. Reconstructed PSNR of frame f_1 obtained by decoding different levels of spatial breakpoints at f_1: The red curve is obtained by decoding all spatial breaks; the green curve shows the results if *no* spatial breakpoints are decoded at f_1, and hence relying on temporal breakpoint induction; the blue thicker curve shows the R-D performance if breakpoints are scalably decoded with respect to the quality of the motion field. (a) Baseball. (b) Space.

In order to evaluate the proposed HST-BPI, we perform one level of temporal decomposition followed by 5 levels of spatial breakpoint-adaptive DWT of the texture and motion data; the wavelet coefficients are then coded using EBCOT [25]. Breakpoints are estimated based on the motion fields, and coded as explained in [21]. The scalable bit-stream is decoded at various quality levels. Fig. 11 shows the reconstructed Y-PSNR of the target frame f_1, and the horizontal axis shows the cost of coding the breakpoints and the texture residual at frame f_1. The precision of the breakpoints at the reference frames f_0 and f_2 is kept high; for the target frame, we either code all spatial vertices (red curve), code no spatial vertices (green curve), or quantize the breakpoints in accordance to the quality associated with the motion field as explained in Sect. VII-B (blue curve).

As expected, the graphs show that at lower bit-rates, where the cost of coding breakpoints becomes significant, the temporal breakpoint induction leads to a significant improvement in R-D performance. There are many complex dependencies between texture, motion, and texture, which are not all accounted for by the analytical model presented in Sect VII.

TABLE II

BD-PSNR AND BD-RATE GAINS OF THE PROPOSED BIHA SCHEME COMPARED TO THE TRADITIONAL (TRAD) ANCHORING. BACKGROUND (BG) AND FOREGROUND (FG) MOTION ACTIVITY IS SUMMARIZED AS STILL, TRANSLATION, ACCELERATION, ROTATION, AND ZOOM

Sequence	Activity BG	Activity FG	Resolution	BD-PSNR	BD-Rate
Baseball	A	1A,1RA	640 × 480	3.35dB	-34.84%
Beach	S	1ZA,2A	640 × 480	0.97dB	-12.46%
Space	S	3A	640 × 480	1.86dB	-24.17%
Winter	S	5A	640 × 480	1.41dB	-14.55%
Autumn	A	2RT,1ZRT	1280 × 736	0.78dB	-9.82%
Butterfly	T	1ZA	1280 × 736	1.58dB	-18.41%
Flowers	R	1RT	1280 × 736	-0.34dB	5.27%
Robots	A	2ZA	1280 × 736	0.63dB	-12.71%
Balls	A	3RA	1920 × 1088	0.82dB	-11.72%
Average	-	-	-	1.23dB	-14.82%

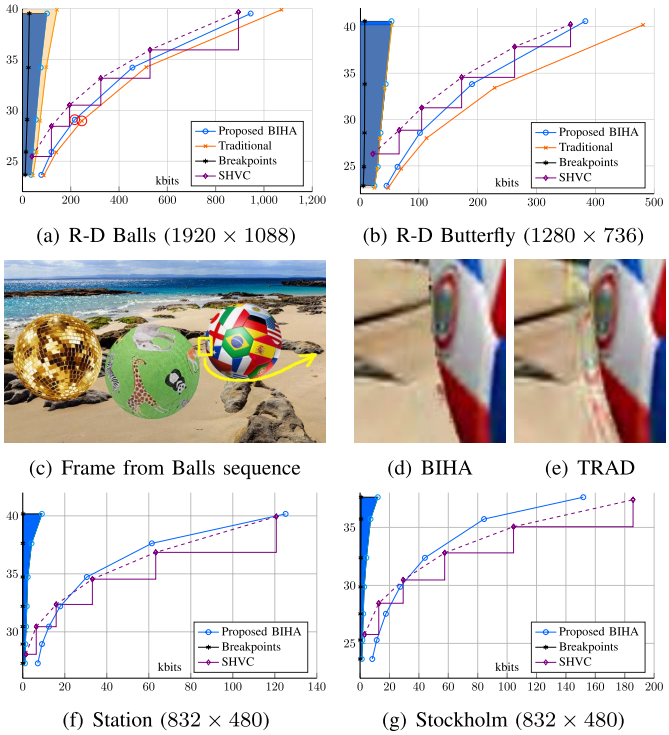
Nonetheless, the resulting scalable rate-allocation leads to a very good R-D performance at all bit-rates; this is evidenced by the blue curve, which closely follows the green curve at low, and the red curve at high bit-rates.

C. Rate-Distortion Results

1) Experimental Setup: For both the proposed BIHA and the TRAD anchoring scheme, the sample sequences are compressed using $T = 3$ levels of temporal decomposition, resulting in 8 frames per "group of pictures" (GOP). The temporal subband frame textures, as well as the differentially coded motion fields, are then subjected to $D = 5$ levels of spatial, breakpoint-adaptive (BPA) DWT. The quantized wavelet coefficients are coded using EBCOT [25]. Breakpoints are coded using the method described in [21], and quantized based on the quality of the motion fields they are coding; intuitively, the more quantized the motion fields, the less breakpoints there are. We remind the reader that all elements of the coded representation are highly scalable. The results are obtained by weighting motion and texture subbands according to (21) and (22), respectively; appropriate weights were also computed for the traditional anchoring scheme.

For SHVC (SHM version 7.0 (HM-15.0)), we used the main profile of the random access encoder (hierarchical B-frames), and created a base layer at QP 38 at half the native resolution of the input sequence, and 5 enhancement layers at full resolution at QPs {23, 26, 30, 34, 38}.

2) BD-PSNR and BD-Rate: We evaluate the rate-distortion performance of the proposed BIHA scheme, and compare it with the traditional way (TRAD) of anchoring motion fields at target frames. This comparison provides a good way of analyzing the benefits of the proposed scheme. Table II compares the R-D performance in terms of BD-PSNR and BD-Rate between the proposed BIHA and the TRAD scheme. One can see that the BIHA scheme outperforms the TRAD scheme in 8 of the 9 sequences, with an average BD-Rate of -14.8%. The bitrate saving on just the motion fields is -13.2%, which shows the effectiveness of the proposed method in terms of predicting motion. We note that the better R-D performance of the BIHA scheme is not solely due to the lower cost of coding the motion, but also because our



(a) R-D Balls (1920 × 1088) (b) R-D Butterfly (1280 × 736)

(c) Frame from Balls sequence (d) BIHA (e) TRAD

(f) Station (832 × 480) (g) Stockholm (832 × 480)

Fig. 12. Average per-frame bit-rate against PSNR for various scenes, obtained using $T = 3$ temporal decomposition levels (GOP size=8). The filled regions on the left show the average bit-rate spent on coding motion field data for the BIHA and TRAD schemes. (c) shows one frame of the balls sequence; (d) and (e) are crops of (c), reconstructed at medium bitrate (red circles in a), for the BIHA and TRAD scheme, respectively. Note how the proposed BIHA scheme has much less ghosting artefacts than the traditional anchoring. (f) and (g) compare the performance of the BIHA scheme with SHVC on two common test sequences, where the motion obtained using the optical flow estimator from [6] almost satisfies the hierarchical consistency required by the proposed method.

scheme is able to produce *geometrically consistent* predictions, even using quantized motion (see Fig. 12d/e for an example); on average, the BD-rate on just the texture data is -17.5%. One can see that the BIHA scheme performs worse in the "Flowers" sequence. This sequence, containing significant rotation of the background, is particularly affected by a shortcoming of the proposed background motion extrapolation technique in disoccluded regions. The problem is that we currently extrapolate background motion for each triangle individually, which creates artificial boundaries in the disoccluded region, which are expensive to code. In future work, we plan to address this issue by performing the background extrapolation on connected disoccluded regions rather than individual triangles, which will avoid such artificial boundaries. Nonetheless, even without such improvement, the proposed scheme clearly outperforms the TRAD method on balance.

3) R-D Comparisons With SHVC: To show the potential and real-world application of the proposed scheme, we provide preliminary comparisons of our method with SHVC. In SHVC, the base and enhancement layers have to be defined at the encoding stage; this allows only for discrete levels of scalability. In contrast, the scalable bitstream created in the BIHA and TRAD schemes can be truncated at any bitrate, allowing for a *highly* scalable framework.

For this reason, BD-Rate/PSNR comparisons between these very different schemes are not very meaningful. Instead, we provide RD-curves for two synthetic, as well as two natural test sequences[4] in Fig. 12; the limited salability of SHCV is indicated by the staircase curve in the figure.

The distortion is expressed in terms of average Y-PSNR for the whole GOP, and the rate on the horizontal axis corresponds to the average number of kbits per frame decoded from the scalable bit-stream. For the BIHA and TRAD schemes, the shaded areas on the left show the average number of bits spent on just the motion fields, and the black curve shows the cost of coding breakpoints (almost identical in the two schemes). One can observe that the R-D performance of the BIHA scheme is approaching the one of SHVC at higher bitrates, and even outperforms SHVC on the two natural sequences; at lower bitrates, SHVC performs better. We emphasize that we are comparing a mature codec with a scheme that has much potential for optimization; better coding of the wavelet coefficients of the motion fields and texture data,[5] would shift the curve of the BIHA (and TRAD) scheme further to the left. Even so, our highly scalable algorithm is competitive with SHVC at higher bitrates.

D. A Few Notes on Complexity

In addition to traditional motion-compensated frame interpolation (MCFI) as found in most video coders, the proposed method involves the following two main steps: 1) transferring breakpoints from reference to target frames, and 2) transferring motion fields from one frame to another.

The proposed method is based on the fact that the underlying motion of a scene can be represented as being piece-wise smooth, with sharp transitions at motion field boundaries. For such motion fields, breakpoints can be expected to be sparse; since the complexity of the breakpoint warping procedure is linear in the number of breakpoints, the computational overhead of transferring breakpoints should not be high.

The core of both motion inversion and motion inference is the cellular affine warping (CAW) procedure, which maps *motion* from one frame to another. For both the horizontal and vertical motion component, the complexity of this motion mapping is similar to the one of MCFI. In the current implementation, the cell size is 1×1 pixels; hence, there are roughly twice as many triangular cells as there are pixels in the video. However, most of the triangles belong to smooth regions with consistent affine models. Similar to the quad-tree structure employed in modern hybrid video coders, a more efficient implementation would use a *hierarchical cell* structure. Whenever a cell contains a nonzero motion wavelet coefficient or a breakpoint, it is split up into 4 smaller cells, until the (sub)cell is smooth and free of breakpoints. If we let the maximum cell size be 32×32, then in the worst case, one nonzero wavelet coefficient or breakpoint creates 5×4 cells. Normally, such nonzero wavelet coefficients are grouped together around moving object boundaries, and multiple coefficients cause the same cell to be further partitioned. In the case of a truly isolated motion coefficient, the associated coding cost can be expected to be high. Assuming 10 bits to code this motion coefficient, the maximum number of partitioned cells per coded motion bit would be $\frac{5\times4}{10} = 2$. The number of cells N_c to be expected is thus linked to the motion bitrate r_m; in practice, we expect $N_c \ll r_m \times 2$. Even this conservative bound suggests typical cell sizes to involve many pixels at reasonable motion bitrates.

[4]Station contains zooming as well as moving objects; Stockholm is dominated by a rotating camera motion, and contains various moving cars.

[5]Each motion field (residual) of each frame at each scale is currently treated as an independent image.

IX. Conclusions and Future Work

We present a novel paradigm for anchoring motion fields employed in video compression. The proposed bidirectional hierarchical anchoring of motion fields (BIHA) in reference frames has some major advantages compared to the traditional way of anchoring motion fields in target frames. Every motion field involved in motion-compensated temporal prediction is warped from reference to target frames – a process during which we observe important properties of the motion field, and obtain a disocclusion mask; this valuable information traditionally has to be communicated as side-information. Furthermore, the motion fields we compute for temporal prediction warp texture in a *geometrically consistent* manner, even if the motion is quantized due to scalable decoding.

The analytical model developed in this paper provides insight into the relative importance and hence the weights to be assigned to the different spatio-temporal texture and motion field subbands. To further improve the scalability attributes of the BIHA scheme, we propose a hierarchical spatio-temporal breakpoint induction (HST-BPI) scheme to allow the transfer of breakpoints from coarse to fine spatio-temporal levels.

Future work includes the development of a joint motion and breakpoint estimation scheme that is optimized for the proposed scheme. We also plan to develop blur models to create more credible transitions around moving object boundaries.

References

[1] J. Jain and A. Jain, "Displacement measurement and its application in interframe image coding," *IEEE Trans. Commun.*, vol. 29, no. 12, pp. 1799–1808, Dec. 1981.

[2] G. J. Sullivan, J. Ohm, W.-J. Han, and T. Wiegand, "Overview of the high efficiency video coding (HEVC) standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 12, pp. 1649–1668, Dec. 2012.

[3] A. Glantz, A. Krutz, and T. Sikora, "Adaptive global motion temporal prediction for video coding," in *Proc. Picture Coding Symp.*, Dec. 2010, pp. 202–205.

[4] R. Mathew and D. S. Taubman, "Quad-tree motion modeling with leaf merging," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 20, no. 10, pp. 1331–1345, Oct. 2010.

[5] M. Tok, V. Eiselein, and T. Sikora, "Motion modeling for motion vector coding in HEVC," in *Proc. Picture Coding Symp.*, May/Jun. 2015, pp. 154–158.

[6] L. Xu, J. Jia, and Y. Matsushita, "Motion detail preserving optical flow estimation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 9, pp. 1744–1757, Sep. 2012.

[7] J. Wulff and M. J. Black, "Modeling blurred video with layers," in *Proc. 13th Eur. Conf.*, vol. 8694. 2014, pp. 236–252.

[8] G. Ottaviano and P. Kohli, "Compressible motion fields," in *Proc. IEEE Conf. Comp. Vis. Pattern Recognit.*, Jun. 2013, pp. 2251–2258.

[9] S. I. Young, R. K. Mathew, and D. S. Taubman, "Joint estimation of motion and arc breakpoints for scalable compression," in *Proc. IEEE Global Conf. Signal Inf. Process.*, Dec. 2013, pp. 479–482.

[10] S. I. Young, R. K. Mathew, and D. S. Taubman, "Embedded coding of optical flow fields for scalable video compression," in *Proc. IEEE 16th Int. Workshop Multimedia Signal Process.*, Sep. 2014, pp. 1–6.
[11] H. Schwarz, D. Marpe, and T. Wiegand, "Overview of the scalable video coding extension of the H.264/AVC standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 9, pp. 1103–1120, Sep. 2007.
[12] P. Helle *et al.*, "A scalable video coding extension of HEVC," in *Proc. IEEE Data Compress. Conf.*, Mar. 2013, pp. 201–210.
[13] A. Secker and D. Taubman, "Motion-compensated highly scalable video compression using an adaptive 3D wavelet transform based on lifting," in *Proc. IEEE Int. Conf. Image Process.*, Oct. 2001, pp. 1029–1032.
[14] B. Pesquet-Popescu and V. Bottreau, "Three-dimensional lifting schemes for motion compensated video compression," in *Proc. IEEE Int. Conf. Acoust. Speech, Signal Process.*, 2001, pp. 1793–1796.
[15] A. Secker and D. Taubman, "Lifting-based invertible motion adaptive transform (LIMAT) framework for highly scalable video compression," *IEEE Trans. Image Process.*, vol. 12, no. 12, pp. 1530–1542, Dec. 2003.
[16] Y. Andreopoulos, A. Munteanu, J. Barbarien, M. Van der Schaar, J. Cornelis, and P. Schelkens, "In-band motion compensated temporal filtering," *Signal Process., Image Commun.*, vol. 19, no. 7, pp. 653–673, Aug. 2004.
[17] N. Mehrseresht and D. Taubman, "An efficient content-adaptive motion-compensated 3-D DWT with enhanced spatial and temporal scalability," *IEEE Trans. Image Process.*, vol. 15, no. 6, pp. 1397–1412, Jun. 2006.
[18] N. Adami, A. Signoroni, and R. Leonardi, "State-of-the-art and trends in scalable video compression with wavelet-based approaches," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 9, pp. 1238–1255, Sep. 2007.
[19] H. G. Lalgudi, M. W. Marcellin, A. Bilgin, H. Oh, and M. S. Nadar, "View compensated compression of volume rendered images for remote visualization," *IEEE Trans. Image Process.*, vol. 18, no. 7, pp. 1501–1511, Jul. 2009.
[20] J.-U. Garbas, B. Pesquet-Popescu, and A. Kaup, "Methods and tools for wavelet-based scalable multiview video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 21, no. 2, pp. 113–126, Feb. 2011.
[21] R. Mathew, D. Taubman, and P. Zanuttigh, "Scalable coding of depth maps with R-D optimized embedding," *IEEE Trans. Image Process.*, vol. 22, no. 5, pp. 1982–1995, May 2013.
[22] D. Rüfenacht, R. Mathew, and D. Taubman, "Hierarchical anchoring of motion fields for fully scalable video coding," in *Proc. IEEE Int. Conf. Image Process.*, Oct. 2014, pp. 3180–3184.
[23] D. Rüfenacht, R. Mathew, and D. Taubman, "Bidirectional hierarchical anchoring of motion fields for scalable video coding," in *Proc. IEEE 16th Int. Workshop Multimedia Signal Process.*, Sep. 2014, pp. 1–6.
[24] A. Mavlankar, S.-E. Han, C.-L. Chang, and B. Girod, "A new update step for reduction of PSNR fluctuations in motion-compensated lifted wavelet video coding," in *Proc. IEEE 7th Int. Workshop Multimedia Signal Process.*, Oct./Nov. 2005, pp. 1–4.
[25] D. Taubman, "High performance scalable image compression with EBCOT," *IEEE Trans. Image Process.*, vol. 9, no. 7, pp. 1158–1170, Jul. 2000.
[26] D. Rüfenacht, R. Mathew, and D. Taubman, "Bidirectional, occlusion-aware temporal frame interpolation in a highly scalable video setting," in *Proc. Picture Coding Symp.*, May/Jun. 2015, pp. 5–9.

Dominic Rüfenacht received the B.Sc. degree in communication systems and the M.Sc. degree in communication systems with a specialization in signals, images, and interfaces from the Swiss Federal Institute of Technology in Lausanne (EPFL), in 2009 and 2011, respectively. He is currently pursuing the Ph.D. degree in electrical engineering with the University of New South Wales, Sydney, Australia. During his undergraduate studies, he was an Exchange Student with the University of Waterloo, ON, Canada, and did his master's thesis entitled Stereoscopic High Dynamic Range Video at Philips Consumer Lifestyle, Eindhoven, Netherlands. From 2011 to 2013, he was with the Image and Visual Representation Group, EPFL, as a Research Engineer, where he was involved in computational photography problems, with an emphasis on color and near-infrared imaging. His research interests are both in computational photography and highly scalable image and video coding.

Reji Mathew received the B.E. degree from the University of Western Australia, Perth, Australia, in 1990, and the M.E. and Ph.D. degrees from the University of New South Wales (UNSW), Australia, in 1996 and 2010, respectively. He was with UNSW Canberra at the Australian Defence Force Academy, from 1996 to 1997, Motorola Laboratories, Motorola Australian Research Centre, Sydney, from 1997 to 2003, and National ICT Australia, Sydney, from 2004 to 2005. He is currently with UNSW, where he pursues his research interests in image and video coding, motion estimation, and scalable representations of motion and depth data.

David Taubman received B.S. and B.E. degrees in electrical engineering from the University of Sydney, in 1986 and 1988, respectively, and the M.S. and Ph.D. degrees from the University of California at Berkeley, in 1992 and 1994, respectively. From 1994 to 1998, he was with Hewlett-Packard's Research Laboratories, Palo Alto, CA. He joined the University of New South Wales in 1998, where he is currently a Professor with the School of Electrical Engineering and Telecommunications. He has authored the book *JPEG2000: Image Compression Fundamentals, Standards and Practice*, with M. Marcellin. His research interests include highly scalable image and video compression, motion estimation and modeling, inverse problems in imaging, perceptual modeling, and multimedia distribution systems. He received the University Medal from the University of Sydney. He has received two best paper awards from the IEEE Circuits and Systems Society for the 1996 paper entitled A Common Framework for Rate and Distortion-Based Scaling of Highly Scalable Compressed Video, and from the IEEE Signal Processing Society for the 2000 paper entitled High Performance Scalable Image Compression with EBCOT.