# Boundary-Control Vector (BCV) Motion Field Representation and Estimation By Using A Markov Random Field Model *

**Ms. No.: F001 -Hs01, (Revised)**

Jin Li † and Xinggang Lin ‡ and C.-C. Jay Kuo †

September 4, 1995

**Abstract**

A new motion field representation based on the boundary-control vector (BCV) scheme for video coding is examined in this work. With this scheme, the motion field is characterized by a set of control vectors and boundary functions. The control vectors are associated with the center points of blocks to control the overall motion behavior. We use the boundary functions to specify the continuity of the motion field across adjacent blocks. For BCV-based motion field estimation, an optimization framework based on the Markov random field model and maximum *a posterior* (MAP) criterion is used. The new scheme effectively represents complex motions such as translation, rotation, zooming and deformation and does not require complex scene analysis. Compared with MPEG of similar decoded SNR (signal-to-noise ratio) quality, 15-65% bit rate saving can be achieved in the proposed scheme with a more pleasant visual quality.

**Keywords:** motion estimation, motion representation, video coding, boundary-control vector (BCV), Markov random field.

## 1 Introduction

In image sequence coding, image frames are often encoded by two parts, i.e. the motion field which represents the change in the sequence and the displacement frame difference (DFD) which represents the residual error after motion compensation. Since both the motion field and DFD have to be transmitted to the receiver, a well designed video coder should balance the bits used in these two parts. Other factors of consideration in video coder design include computational cost, hardware complexity and the domain of applicability.

*This work was supported in part by the Chinese National Science Foundation Grant No. 69272003, in part by the U.S. National Science Foundation Presidential Faculty Fellow (PFF) Award ASC-9350309 and in part by the Rockwell Internationl Corporation.

†The authors are with the Signal and Image Processing Institute and the Department of Electrical Engineering-Systems, University of Southern California, Los Angeles, California 90089-2564.E-mail: lijin@sipi.usc.edu and cckuo@sipi.usc.edu.

‡The author is with the Image Processing Division and the Department of Electronic Engineering, Tsinghua University, Beijing, 100084, P.R.China, E-mail: deelxg@sunstation2.tsinghua.edu.cn.

We can roughly classify existing motion field representations into block-based, pel-based and model-based three categories. The block-based representation has been widely used and adopted by several standards such as H.261 [16] and MPEG [6]. It divides an image frame into nonoverlapping blocks, and represents the motion field in each block with a translation vector. This representation is generally applicable and concise. A differential coding can be used to further reduce the redundancy between motion vectors by exploiting their spatial correlation. The block-based motion field can be estimated by using a straightforward block matching algorithm (BMA) or its variants. However, the block-based scheme has some limitations. It does not well represent complicated motion types such as rotation, zooming and deformation, neither does it give any considerations to the motion boundaries of moving objects. The block effect caused by motion discontinuity between two adjacent blocks is subjectively annoying. The poor quality of the motion compensated image calls for more bits to encode the DFD so that the total bit rate increases. Research has been performed to overcome the above shortcomings. Orchard [17] incorporated motion discontinuity into the block-based motion representation to obtain substantial quality improvement along moving object boundaries. The discontinuity between the motion vectors in adjacent blocks however still exists. Bergeron [1], Fuh [4], Papadopoulos [18] and Seferidis [20], [21] used more complex functions such as affine, perspective, bilinear, 2nd-order polynomial transformations to represent the motion field inside each block. The schemes require more bits in the coding of motion field. As an example, an affine representation requires 3 vectors, instead of the conventional 1 vector, to represent the motion field in a block.

For the pel-based representation, each pixel has its own motion vector. An arbitrary motion field can be easily represented by this scheme. The tradeoff is that it requires a lot of bits in representing such a dense motion field. To avoid the transmission of the bits for the dense motion field, one approach [15] is to derive the motion field of the current frame from that of the previous one in the encoder as well as the decoder. Since the decoder contains a complicated motion estimation unit, its cost and decoding time become issues. Furthermore, since not all motion in a video sequence can be well predicted and noise in the scenery can greatly influence the result of the estimation algorithm, the predicted gain is usually low. Due to these reasons, the pel-based motion estimation approach is not widely used now.

The model-based approach [2, 7, 14, 25] has received a lot of attention recently. Scenery is segmented into objects and the background by using image analysis tools in encoding. The shape, texture and motion information of objects are then transmitted. Image scenery is regenerated from the transmitted shape, texture and motion information with the computer graphic technique at

the decoder. The model-based coding scheme has two major advantages, i.e. high compression efficiency and visually insensitive distortion. Since the shape and texture of objects are relatively steady across multiple frames, we only have to transmit a small amount of motion information so that the compression efficiency can be high. When the error occurs in estimation or representation, the distortion leads to small changes in the shape (geometrical deformation), texture and position of the object, which is less sensitive to human eyes in comparison with the block effect. The main shortcoming of the model-based scheme is that it requires complex scene analysis which is very expensive to implement. Coding scenery is usually restricted (e.g. the head-and-shoulder video sequence) so that the image analysis can be simplified. The restriction greatly limits the applicability of the model-based approach.

In this research, we develop a novel motion field representation scheme in which the motion field is characterized by boundaries and control vectors on a predefined grid points. In contrast with the block-based motion representation, the control vector does not represent the motion field for every pixel within the block but only for the control point. The motion vector at points other than control points are obtained via interpolation. The boundary function in the BCV-scheme specifies the discontinuity of the motion field so that the unrealistic global smooth constraint can be removed. The new boundary-control vector (BCV) scheme can describe many different complex motions such as translation, rotation, zooming and deformation and is applicable to a wide variety of scenes with a very concise representation. Compared with the model-based scheme, the BCV scheme does not require complicated scenery analysis. For more details on the BCV motion field representation and the comparison of this new scheme with other traditional representations, we refer to the discussion in Sections 2.2 and 2.3.

This paper is organized as follows. The boundary-control vector (BCV) motion field representation is introduced in Section 2. A framework is proposed to estimate the BCV motion field based on the Markov random field model. The coding of predicted error after BCV motion compensaton is discussed in Section 4. Experimental results are given in Section 5 to demonstrate the performance of our proposed method. Concluding remarks are presented in Section 6.

# 2 Boundary-Control Vector (BCV) Motion Field Representation

## 2.1 Control Vectors and Boundaries

Consider an image of size $N \times N$ which can be divided into $B \times B$ nonoverlopping blocks of size $K \times K$ (i.e. $N = B \times K$). We choose the center of each block as the control point so that the position of each control point can be expressed as

$$\Psi(a,b) = (aK + \frac{K+1}{2}, bK + \frac{K+1}{2}), \quad 0 \leq a,b \leq B-1, \tag{2.1}$$

where, without loss of generality, $K$ is assumed to be an even number. These control points are fixed throughout the entire image sequence. We use $\mathbf{\Psi}$ to denote the set of all control points. The motion vector at control point $\Psi(a,b)$ at time $t$ is denoted by $\mathbf{D}(a,b,t)$. It is simply called the *control vector*. We use $\mathbf{D_t}$ to denote the set of all control vectors at time $t$. Along the lower and right boundaries of the block centered at $\Psi(a,b)$, we define, respectively, binary functions $E_h(a,b,t)$ and $E_v(a,b,t)$ whose value is 1 if there is a discontinuity in the motion field along the corresponding boundary. Otherwise, it is 0. We denote the location of the boundary as $\Phi_h(a,b)$ and $\Phi_v(a,b)$. For convenience, we say a boundary exists if its value is 1. We use the vector notation $\mathbf{E}(a,b,t)$ to denote the boundary pair $(E_h(a,b,t), E_v(a,b,t))$ and $\mathbf{E_t}$ is the set of all boundary functions at time $t$. Correspondingly, We use $\mathbf{\Phi}$ to denote the set of all boundaries. An illustration of control points and boundaries is given in Fig. 1.

## 2.2 BCV Motion Field

In BCV motion field representation, the control vector $\mathbf{D}(a,b,t)$ does not represent the motion field for the whole block centered at $(a,b)$ but only for the control point $\Psi(a,b)$. The motion vector at points other than control points are obtained via *interpolation*. It was however observed [22] that most of the motion compensated errors occur around the boundaries of moving objects with the block-based method, the DFD around moving object boundaries and the DFD inside the object can behave quite differently. In this research, we introduce the motion field boundary, so that we can locate motion discontinuity and treat the two kinds of DFD separately. The unrealistic global smooth constraint of the motion field can therefore be removed to improve the overall video coding efficiency.

In BCV scheme, the individual motion vector of each pixel is interpolated from the control vector and the motion boundary. The interpolation process is classified into 4 different modes,

Figure 1: Illustration of control points and boundaries with $K = 4$, where the empty circle denotes the pixel position.

which are summarized in Fig. 2. When no boundary exists among four neighboring control points, a bilinear interpolation is applied. It is called the basic mode. When boundaries exist, some motion field discontinuity is present and the four neighboring control points can be divided into 2-2 or 3-1 pairs as shown in Fig. 2. The 2-2 pair case is called mode B, and a linear interpolation is performed at the pixel of interest by using the two control vectors associated with the region. The 3-1 pair case can be further classified into modes A1, A2 and C. If there is only one control vector associated with the region, the case is called mode C, in which all pixels inside take the same constant value. If the pixel is surronded by three adjacent control vectors, the case is called mode A1, in which linear interpolation is performed by using the three associated control vectors. Finally, if the pixel is associated by three control vectors, but it is outside the triangle formed by the three control vectors, the case is called mode A2, in which we adopt a simple linear weighting scheme. In developing the above interpolation rules, we want to keep them as simple as possible so that the motion vector of each pixel can be computed effectively and, in the mean time, we ensure that the discontinuity of the motion field only occurs at motion boundary.

An example of the interpolated motion field is shown in Fig. 3. We denote the interpolation

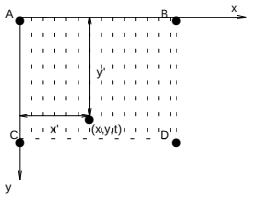
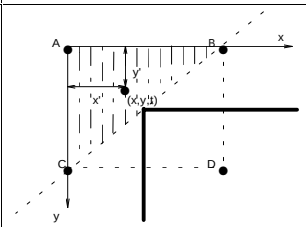
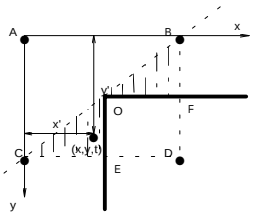
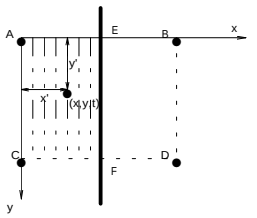
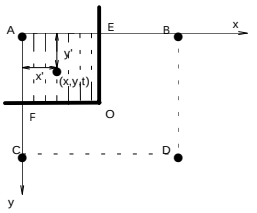
| Mode | Figure | Interpolation region | Interpolation Scheme |
|---|---|---|---|
| Basic |  | □ABCD | $d(x,y,t) = D_A + \frac{x'}{K}(D_B - D_A) + \frac{y'}{K}(D_C - D_A) + \frac{x'y'}{K^2}(D_A + D_D - D_B - D_C)$ (B.1) |
| A1 |  | ΔABC | $d(x,y,t) = D_A + \frac{x'}{K}(D_B - D_A) + \frac{y'}{K}(D_C - D_A)$ (B.2) |
| A2 |  | ΔBOFU ΔCOE | $d(x,y,t) = \frac{K - y'}{2K - x' - y'} D_B + \frac{K - x'}{2K - x' - y'} D_C$ (B.3) |
| B |  | □AEFC | $d(x,y,t) = \frac{y'}{K} D_C + \frac{K - y'}{K} D_A$ (B.4) |
| C |  | □AEOF | $d(x,y,t) = D_A$ (B.5) |

Figure 2: Summary of interpolation rules for BCV motion field representation, where $A$, $B$, $C$ and $D$ denote four control points, $D_A$, $D_B$, $D_C$ and $D_D$ are the associated control vectors, and a solid line indicates the presence of boundary discontinuity.

Figure 3: An example of BCV motion field interpolation where different colors represent different interpolation modes.

process of the dense motion field by

$$\mathbf{d}(x, y, t) = \text{gen}\{\mathbf{D_t}, \mathbf{E_t}\}, \tag{2.2}$$

A motion field is uniquely specified by the sets $\mathbf{D_t}$ and $\mathbf{E_t}$ through (2.2).

## 2.3 Comparison of Motion Field Representations

It is worthwhile to comment on the unique features of the proposed BCV-scheme in comparison with the three other motion field representations, i.e. pel-based, block-based and model-based representations.

With respect to the pel-based motion representation, the motion boundary of the BCV scheme is relatively coarse, i.e. only represented in block accuracy. This is a tradeoff between the predicted gain that can be achieved and the bits required to transmit the boundary information. For example, with pel-accurate boundaries, we still cannot describe all interframe changes, say, the exposure region. The estimation from the pel-based approach can be easily affected by noise, and the estimation error in the object boundary can cause degradation in the displacement frame. Eventhough the distortion is restricted to a very small region, it is visually annoying. Pieces of background may move with the object, and pieces of the object may remain on the background. In the BCV

scheme, we compromise the bits between DFD coding and the boundary representation. We use the coarse-level boundaries and admit that there may exist some significant DFD around boundaries in the displacement frame. Although more bits are required to encode the DFD, the bits used to encode the motion information can be greatly saved.

The motion field in the block-based representation is constant over each block and in general not continuous along the block boundary. In contrast, the BCV-based motion field is continuous if there is no boundary between neighboring control points. The continuity of the motion field removes the visually annoying block effect in the displacement frame. Although the bilinear interpolation only provides an approximation of a complex motion field and the control vectors may have errors, the derivative object in the displacement frame only deforms slightly. Human eyes are less sensitive to these smooth errors. Along the object boundary, since only coarse boundary is used in BCV, the block effect remains and the DFD in these regions is large. However, boundary regions contstitute only a small portion of the whole scenery so that the coding efficiency of the BCV scheme is still higher. More importantly, the BCV-based motion field can represent a large class of complicated motions more accurately. The translation and rotation of objects formed by 3-D planar surfaces can be well described by BCV. For the more complicated object movement, the bilinear interpolation scheme used by BCV serves as a good first order approximation of the true motion field. The BCV-based motion field can be stored or transmitted almost as efficiently as the block-based motion field. In addition to control vectors, only a small number of bits are required to encode the boundary information. To illustrate the visual difference by using the BCV and block-based methods, we show two coded displacement frames for the TREVOR image sequence in Fig. 4. Even though the the displacement frames only differ by 1.77 dB in Fig. 4, the subjective quality of the BCV-based displacement frame is much better. There is a very visible block effect in the block-based method, e.g. the region of the right hand and fingers. In contrast, the BCV-based displacement frame is smoother with no visible block effect.

The BCV motion field representation is quite different from the object or model-based coding scheme, in which objects in the scene have to be analyzed first. No complex image analysis is required by BCV, and no *a prior* knowledge about the image scenery is needed. The BCV scheme is applicable to a wide variety of scenes. A related video coding scheme known as the active mesh [24] uses a deformable mesh structure to describe the motion field, where complex image analysis is required to track the change of the structure, say, the merge and creation of the mesh. With a set of position invariant control vectors in BCV, the motion estimation task can be greatly simplified.

Figure 4(a)

(b)

Figure 4: Comparison of displacement frames with (a) block-based scheme (0.0186bpp, 29.80dB) and (b) BCV scheme (0.0159bpp, 31.57dB).

# 3 Estimation of Control Vectors and Boundary Functions

## 3.1 Problem Formulation

In this section, we focus on the estimation problem for the BCV represenation scheme presented in the previous section. The parameters to be estimated include the control vectors and the boundary functions. We use

$$\mathbf{f_t} = \{f(x,y,t) \mid 0 \le x < N,\ \ 0 \le y < N,\ \ x,y \in \mathbf{I}\} \tag{3.1}$$

to denote the image of size $N \times N$ at time instance $t$. Let $\Delta t$ be the time interval between two successive frames. Then, the estimation problem can be stated mathematically as: given $\mathbf{f_t}$ and $\mathbf{f_{t-\Delta t}}$, we want to determine the set $\mathbf{D_t}$ of control vectors and the set $\mathbf{E_t}$ of boundary functions. Once $\mathbf{D_t}$ and $\mathbf{E_t}$ are specified, the motion vector at every pixel can be uniquely determined via interpolation.

To calculate the BCV-based motion field, the following maximum *a posterior* (MAP) criterion [19] is considered:

$$P(\mathbf{E_t^*}, \mathbf{D_t^*} \mid \mathbf{f_t}, \mathbf{f_{t-\Delta t}}) \equiv \max_{\mathbf{E_t}, \mathbf{D_t}} P(\mathbf{E_t}, \mathbf{D_t} \mid \mathbf{f_t}, \mathbf{f_{t-\Delta t}}). \tag{3.2}$$

By applying the Bayes rule [11], we have

$$P(\mathbf{E_t}, \mathbf{D_t} \mid \mathbf{f_t}, \mathbf{f_{t-\Delta t}}) = \frac{P(\mathbf{f_{t-\Delta t}} \mid \mathbf{E_t}, \mathbf{D_t}, \mathbf{f_t}) P(\mathbf{E_t}, \mathbf{D_t} \mid \mathbf{f_t})}{P(\mathbf{f_{t-\Delta t}} \mid \mathbf{f_t})}. \tag{3.3}$$

The denomenator term $P(\mathbf{f_{t-\Delta t}} \mid \mathbf{f_t})$ is independent of $\mathbf{E_t}$ and $\mathbf{D_t}$ and can be ingored in the optimization procedure. Thus, the above MAP problem is equivalent to the maximization of the numerator

$$P(\mathbf{f_{t-\Delta t}} \mid \mathbf{E_t}, \mathbf{D_t}, \mathbf{f_t}) P(\mathbf{E_t}, \mathbf{D_t} \mid \mathbf{f_t}). \tag{3.4}$$

The $P(\mathbf{f_{t-\Delta t}} \mid \mathbf{E_t}, \mathbf{D_t}, \mathbf{f_t})$ and $P(\mathbf{E_t}, \mathbf{D_t} \mid \mathbf{f_t})$ are related, respectively, to the displacement frame difference and *a prior* distribution of the control vector set $\mathbf{D_t}$ and the motion boundary set $\mathbf{E_t}$ as detailed below.

Let us first focus on the term $P(\mathbf{f_{t-\Delta t}} \mid \mathbf{E_t}, \mathbf{D_t}, \mathbf{f_t})$. When the motion field is generated via

$$\mathbf{d}(x,y,t) = \mathrm{gen}\{\mathbf{E_t}, \mathbf{D_t}\}, \tag{3.5}$$

the displacement frame at time $t$ is obtained by translating the image at $t - \Delta t$, i.e.

$$DF(x,y,t) = f(x - d_x(x,y,t), y - d_y(x,y,t), t - \Delta t), \tag{3.6}$$

and the difference between the frame $\mathbf{f_t}$ and the displacement frame is called the displacement frame difference and denoted by

$$DFD(x,y,t) = f(x,y,t) - DF(x,y,t). \tag{3.7}$$

The vector forms of the displacement frame and displacement frame difference are given by $\mathbf{DF_t}$ and $\mathbf{DFD_t}$, respectively. It is straightforward to derive that

$$\begin{aligned} P(\mathbf{f_{t-\Delta t}} \mid \mathbf{E_t}, \mathbf{D_t}, \mathbf{f_t}) &= P(\mathbf{DF_t} \mid \mathbf{E_t}, \mathbf{D_t}, \mathbf{f_t}) \\ &= P(\mathbf{f_t - DFD_t} \mid \mathbf{E_t}, \mathbf{D_t}, \mathbf{f_t}) \\ &= P(\mathbf{-DFD_t} \mid \mathbf{E_t}, \mathbf{D_t}, \mathbf{f_t}). \end{aligned} \tag{3.8}$$

Empirically speaking, the displacement frame difference $\mathbf{DFD_t}$ is independent of $\mathbf{E_t}$, $\mathbf{D_t}$ and $\mathbf{f_t}$ and can be modeled as a white Gaussian function

$$P(\mathbf{DFD_t}) = (2\pi\sigma^2)^{-\frac{N^2}{2}} \exp\left\{ -\frac{1}{2\sigma^2} \sum_{(x,y)} DFD^2(x,y,t) \right\}, \tag{3.9}$$

where the deviation $\sigma$ can be predicted via

$$\sigma = \sqrt{\frac{1}{N^2} \sum_{(x,y)} DFD^2(x,y,t-\Delta t)}. \tag{3.10}$$

Based on (3.8) and (3.9), we conclude that

$$P(\mathbf{f_{t-\Delta t}} \mid \mathbf{E_t}, \mathbf{D_t}, \mathbf{f_t}) = (2\pi\sigma^2)^{-\frac{N^2}{2}} \exp\left\{ -\frac{1}{2\sigma^2} \sum_{(x,y)} DFD^2(x,y,t) \right\}. \tag{3.11}$$

We use the Markov random field (MRF) model to determine *a prior* probability distribution $P(\mathbf{E_t}, \mathbf{D_t} \mid \mathbf{f_t})$, which is detailed below.

### 3.2 Markov Random Field Model

The Markov random field (MRF) model has been successfully used in image restoration [5], motion detection [23], and motion estimation [3, 10]. In this work, it is used for BCV-based motion field estimation. In this subsection, we give a brief description of the MRF model.

**Definition 1** *Let* $\mathbf{S} = \{s_1, s_2, \cdots, s_N\}$ *be a set of sites and* $\mathbf{G}(s)$ *the set of neighbors of* $s \in \mathbf{S}$. *We call* $\mathcal{G}(\mathbf{S}) = \{\mathbf{G}(s) \mid s \in \mathbf{S}\}$ *a* **neighborhood system** *if the following three conditions are satisfied:*

1. $\mathbf{G}(s) \subset S$,

(a) (b)

Figure 5: The neighborhood system for the BCV scheme: (a) neighborhood system $\mathbf{G}(\Psi(a,b))$ of control point $\Psi(a,b)$ and (b) neighborhood system $\mathbf{G}(\Phi(a,b))$ of boundary $\Phi(a,b)$.

*2.* $s \notin \mathbf{G}(s)$,

*3.* $s \in \mathbf{G}(r) \Longleftrightarrow r \in \mathbf{G}(s)$.

*The pair* $\{\mathbf{S}, \mathcal{G}(\mathbf{S})\}$ *is called a* **graph**.

We assume that every control vector $D(a,b,t)$ and boundary element $E(a,b,t)$ only correlates with others in a small neighborhood for the BCV scheme. A first-order neighborhood system $\mathcal{G}(\mathbf{S})$ is illustrated in Fig. 5, which consists of two parts: the neighborhood system of control point $\mathbf{G}(\Psi(a,b))$ and the neighborhood system of boundary $\mathbf{G}(\Phi(a,b))$. Thus, the control vector set $\mathbf{D_t}$ and the boundary set $\mathbf{E_t}$ can be treated as a coupled MRF with joint sites $\mathbf{S}$:

$$\mathbf{S} = \mathbf{\Psi} \cup \mathbf{\Phi}. \tag{3.12}$$

**Definition 2** *A subset* $\mathbf{C} \subset \mathbf{S}$ *is a* **clique**, *if* $\mathbf{C}$ *satisfies*

$$\forall r, s \in \mathbf{C}, \quad r \neq s \Longrightarrow s \in \mathbf{G}(r). \tag{3.13}$$

All cliques for the BCV scheme are depicted in Fig. 6. We use $\mathcal{C}(\mathbf{S})$ to denote the set of all cliques in $\mathbf{S}$.

Figure 6: All possilbe cliques for the BCV scheme.

**Definition 3** *Let* $X = \{x(s) \mid s \in \mathbf{S}\}$ *denote a family of random variables indexed by* $\mathbf{S}$, $\mathbf{\Omega}_k = \{\omega(s_k) \mid s_k \in \mathbf{S}\}$ *the configuration space of* $x(s_k)$ *and* $\mathbf{\Omega} = \{\omega = (\omega(s_1), \omega(s_2), \cdots, \omega(s_N))\}$ *the configuration space of the random variable set* $X$. *Then,* $\mathbf{X}$ *is a markov random field (MRF) with respect to* $\mathcal{G}(\mathbf{S})$, *if*

1. $P(\mathbf{X} = \omega) > 0 \qquad \forall \omega \in \mathbf{\Omega}$,
2. $P(x(s) = \omega(s) \mid x(r) = \omega(r), r \neq s) = P(x(s) = \omega(s) \mid x(r) = \omega(r), r \in \mathbf{G}(s))$,

*or the probability distribution of random variable* $x(s)$ *is only relevant to its neighborhood* $\mathbf{G}(s)$.

There is an important theorem regarding the probability distribution of the MRF.

**Theorem 1** *The random variable set* $\mathbf{X}$ *is a MRF with respect to* $\mathcal{G}(\mathbf{S})$ *if and only if*

$$P(\mathbf{X} = \omega) = \frac{1}{A} \exp\left\{ - \sum_{\mathbf{C} \in \mathcal{C}(\mathbf{S})} V_C(\omega) \right\}, \tag{3.14}$$

*where each* $V_C(\omega)$ *is a potential function depends only on those* $x(s)$ *for which* $s \in \mathbf{C}$, *and* $A$ *is a normalizing constant so that*

$$\int_{\omega \in \mathbf{\Omega}} P(\mathbf{X} = \omega) d\omega = 1. \tag{3.15}$$

The proof of the theorem can be found in [8].

In the application of MRF, we are often required to determine the probability distribution of $\mathbf{X}$. By using Theorem 1, we can define the probability distribution $P(\mathbf{X} = \omega)$ by designating the potential function $V_C(\omega)$ for each clique $\mathbf{C} \in \mathcal{C}(\mathbf{S})$ given in Fig. 6. Although there are altogether sixteen clique forms in figure 1, some of them can be merged or eliminated. For example, with the observation that the value of control vector $D(a, b, t)$ may only correlate with its neighbor vector, it is irrelevant to the surrounding boundary elements, the potential of clique (p) is just the sum of potential (e) and (a), so (p) can be eliminated, and so do the clique (f),(j),(m),(o). For the remaining cliques, some are subsets of others, we can optionally merge them based on whether they have an independent physical sense. As an example, let us consider clique (n), which is a subset of clique (e). The potential description of clique (n) can be included in the potential description of clique(e), and they are both shape descriptions for the boundary element, so that clique (n) can be merged into clique (e). With the same reason, clique (l) can be merged into clique (e), cliques (h),(i),(k) can be merged into clique (d), and clique (g) can be merged into clique (c). On the other hand, although clique (b) is a subset of clique (c), they have independent physical sense.

That is, clique (b) can be interpreted as the correlation of boundary element with the intensity edges, clique (c) can be interpreded as the correlation between neighbor control vectors. Therefore, we decide to keep them seperate. After carefully merging and eliminating the unnecessary clique forms, we chose five clique forms (a)-(e) for representing the rior distribution of the motion field $P(\mathbf{E_t}, \mathbf{D_t} \mid \mathbf{f_t})$, with each carry its own independent physical sense:

*A.* $\mathbf{C_a} = \{\Psi(a,b)\}$*: the probability distribution of the single control vector* $D(a,b,t)$

The potential of clique (a) is determined by the *a prior* distribution of the single control vector $D(a,b,t)$. If we assume $D(a,b,t)$ to be conformed to gaussian distribution, the potential of clique (a) will be:

$$V_{C_a}(\mathbf{E_t}, \mathbf{D_t}, \mathbf{f_t}) = V_{C_a}(\mathbf{D_t}) = \alpha_a \cdot \|D(a,b,t)\|^2 \tag{3.16}$$

If we assume $D(a,b,t)$ to be conformed to laplacian distribution, the potential of clique (a) will be:

$$V_{C_a}(\mathbf{E_t}, \mathbf{D_t}, \mathbf{f_t}) = V_{C_a}(\mathbf{D_t}) = \alpha_a \cdot \|D(a,b,t)\| \tag{3.17}$$

If we assume $D(a,b,t)$ to be conformed to uniform distribution, the potential of clique (a) will be:

$$V_{C_a}(\mathbf{E_t}, \mathbf{D_t}, \mathbf{f_t}) = V_{C_a}(\mathbf{D_t}) = 0 \tag{3.18}$$

In the paper, we chose the uniform distribution model (3.18).

*B.* $\mathbf{C_b} = \{\Phi(a,b)\}$*: the correlation of the boundary element* $E(a,b,t)$ *with the intensity edges*

We define the potential of clique (b) to be inverse propotional to the intensity edge factor:

$$V_{C_b}(\mathbf{E_t}, \mathbf{D_t}, \mathbf{f_t}) = V_{C_b}(E(a,b,t), \mathbf{f_t}) = \begin{cases} \frac{E_h(a,b,t)}{edge_h(a,b,\mathbf{f_t})} \\ \\ \frac{E_v(a,b,t)}{edge_v(a,b,\mathbf{f_t})} \end{cases} \tag{3.19}$$

$edge_h(a,b,\mathbf{f_t})$ or $edge_v(a,b,\mathbf{f_t})$ measures the degree of discontinuity in the intensity image, it is formularized as:

$$edge_h(a,b,\mathbf{f_t}) = \left| \sum_{(x,y) \in B(a,b) \cup B(a,b-1)} Q[f(x,y,t), f(x,y+1,t)] \right| \tag{3.20}$$

$Q(a,b)$ is the number of large positive or negative transition between pixel $a$ and $b$:

$$Q(a,b) = \begin{cases} \left\lfloor \frac{b-a}{b+a} \cdot T_e \right\rfloor & b > a \\ \\ -\left\lfloor \frac{a-b}{b+a} \cdot T_e \right\rfloor & b < a \end{cases} \tag{3.21}$$

where $\lfloor x \rfloor$ is the maximum integer value that is not greater than $x$.

*C.* $\mathbf{C_c} = \{\Psi(a,b), \Phi_h(a,b), \Psi(a,b-1)\}$ *or* $\{\Psi(a,b), \Phi_v(a,b), \Psi(a-1,b)\}$*: the correlation between neighbor control vectors.*

Note that the boundary element $E(a,b,t)$ between two neighbor control vectors has substantial influence to the correlation of the control vectors. If there is a boundary between the two neighbor control vectors, the two vectors will belong to different interpolation object $O_i$. They will have no correlation. Otherwise the two vectors will belong to the same object, and therefore have strong correlations. We define the potential of clique (c) as:

$$\begin{aligned} V_{C_c}(\mathbf{E_t}, \mathbf{D_t}, \mathbf{f_t}) &= V_{C_c}(\mathbf{E_t}, \mathbf{D_t}) \\ &= \begin{cases} \mid D(a,b,t) - D(a,b-1,t) \mid & E_h(a,b,t) = 0 \\ \mid 0 & E_h(a,b,t) = 1 \end{cases} \quad \Phi_h(a,b) \in \mathbf{C_c} \end{aligned} \tag{3.22}$$

*D.* $\mathbf{C_d} = \{\Phi_h(a,b), \Phi_h(a-1,b), \Phi_v(a,b), \Phi_v(a,b-1)\}$ *and* $\mathbf{C_e} = \{\Phi_h(a,b), \Phi_v(a,b), \Phi_h(a,b+1), \Phi_v(a+1,b)\}$*: the a prior knowledge of the boundary set* $\mathbf{E_t}$

We define the potential of clique (d) and (e) according to *a prior* distribution of the boundary set $\mathbf{E_t}$.

$$V_{C_d}(\mathbf{E_t}, \mathbf{D_t}, \mathbf{f_t}) = V_{C_d}(\mathbf{E_t}) \quad \text{and} \quad V_{C_e}(\mathbf{E_t}, \mathbf{D_t}, \mathbf{f_t}) = V_{C_e}(\mathbf{E_t}). \tag{3.23}$$

The considerations in the definition of the potential include the suppression of boundaries to reduce the number of interpolation regions and the smoothness and closeness of the boundary set $\mathbf{E_t}$ to form a reasonable object. We define the potential for cliques $\mathbf{C_d}$ and $\mathbf{C_e}$ in Figs. 7 and 8.

By using Theorem 1 and the potentials of cliques (a)-(e) defined above, we can obtain *a prior* distribution of the boundary set $\mathbf{E_t}$ and the control vector set $\mathbf{D_t}$ as:

$$P(\mathbf{E_t}, \mathbf{D_t} \mid \mathbf{f_t}) = \frac{1}{A_1} \exp\{-U_1(\mathbf{E_t}, \mathbf{D_t}, \mathbf{f_t})\}, \tag{3.24}$$

where $A_1$ is a normalization factor and

$$\begin{aligned} U_1(\mathbf{E_t}, \mathbf{D_t}, \mathbf{f_t}) &= \sum_{\mathbf{C} \in c(\mathbf{S})} V_C(\mathbf{E_t}, \mathbf{D_t}, \mathbf{f_t}) \\ &= \alpha_b \cdot \sum_{\mathbf{C_b}} V_{C_b}(\mathbf{E_t}, \mathbf{f_t}) + \alpha_c \cdot \sum_{\mathbf{C_c}} V_{C_c}(\mathbf{D_t}, \mathbf{E_t}) + \alpha_d \cdot \sum_{\mathbf{C_d}} V_{C_d}(\mathbf{E_t}) \\ &\quad + \alpha_d \cdot \sum_{\mathbf{C_e}} V_{C_e}(\mathbf{E_t}), \end{aligned} \tag{3.25}$$

and where $\alpha_b$, $\alpha_c$, $\alpha_d$ are the weighting factors for different cliques. The value of $\alpha_b$ relates to the degree of correlation between the motion boundary and the intensity edge, the value of $\alpha_c$ relates

Figure 8: Potential $V_{C_e}$ (For rotational aliases, only one is listed in the figure)

to the degree of correlation between neighbor control vectors, and the value of $\alpha_d$ relates to *a prior* restriction of the boundary set $\mathbf{E_t}$. By substituting (3.8), (3.24), (3.25) in (3.4), we obtain

$$P(\mathbf{f_{t-\Delta t}} \mid \mathbf{E_t}, \mathbf{D_t}, \mathbf{f_t}) P(\mathbf{E_t}, \mathbf{D_t} \mid \mathbf{f_t}) = \frac{1}{A} \exp\{-U(\mathbf{E_t}, \mathbf{D_t}, \mathbf{f_t}, \mathbf{f_{t-\Delta t}})\}, \tag{3.26}$$

where $A$ is a normalization factor and

$$\begin{aligned} U(\mathbf{E_t}, \mathbf{D_t}, \mathbf{f_t}, \mathbf{f_{t-\Delta t}}) &= \frac{1}{2\sigma^2} \sum_{(x,y)\in \mathbf{I}} DFD^2(x,y,t) + \alpha_b \cdot \sum_{\mathbf{C_b}} V_{C_b}(\mathbf{E_t}, \mathbf{f_t}) \\ &+ \alpha_c \cdot \sum_{\mathbf{C_c}} V_{C_c}(\mathbf{E_t}, \mathbf{D_t}) + \alpha_d \cdot [\sum_{\mathbf{C_d}} V_{C_d}(\mathbf{E_t}) + \sum_{\mathbf{C_e}} V_{C_e}(\mathbf{E_t})]. \end{aligned} \tag{3.27}$$

The four terms in (3.27) are called, respectively, the displacement frame difference, the correlation between the motion boundary (motion discontinuity) and the intensity edges, the correlation between neighbor control vectors, and the *a prior* restriction of the boundary set $\mathbf{E_t}$.

### 3.3 Estimation via Optimization

By using the analysis given above, the problem of finding the MAP estimation of the motion field $d^*(x,y,t) = gen\{\mathbf{E_t^*}, \mathbf{D_t^*}\}$ is converted to the determination of the minimum point of the potential $U(\mathbf{E_t}, \mathbf{D_t}, \mathbf{f_t}, \mathbf{f_{t-\Delta t}})$, i.e.

$$U(\mathbf{E_t^*}, \mathbf{D_t^*}, \mathbf{f_t}, \mathbf{f_{t-\Delta t}}) = \min_{\mathbf{E_t}\in\mathcal{E}_t, \mathbf{D_t}\in\mathcal{D}_t} U(\mathbf{E_t}, \mathbf{D_t}, \mathbf{f_t}, \mathbf{f_{t-\Delta t}}), \tag{3.28}$$

where the potential $U(\mathbf{E_t}, \mathbf{D_t}, \mathbf{f_t}, \mathbf{f_{t-\Delta t}})$ is given by (3.27). To obtain the global minimum of the potential function $U(\mathbf{E_t}, \mathbf{D_t}, \mathbf{f_t}, \mathbf{f_{t-\Delta t}})$ is not an easy task. On one hand, since the configuration space $\{\mathcal{E}_t, \mathcal{D}_t\}$ of the BCV motion field is extremely large, it is impactical that we search the whole space for the minimum point. On the other hand, since the potential $U(\mathbf{E_t}, \mathbf{D_t}, \mathbf{f_t}, \mathbf{f_{t-\Delta t}})$ is a highly nonconvex function with many local extrema and the boundary element $E(d,a,b,t)$ takes a binary value, we cannot adopt a gradient based algorithm to find the minimum point. Since the problem formulation involves the MRF, a standard simulated annealing [9] process is used to solve this problem. We update the control vector $D(a,b,t)$ and the boundary element $E(a,b,t)$ once at a time, and a decreasing temperature sequence $T_k$ is used to control the update along the process. For each update, the change in the potential $U$ is evaluated together with the temperature $T_k$ to determine whether the update is accepted or not. In the beginning, the temperature $T_k$ is high so that the update is very random to allow the algorithm to select a good starting point to avoid being trapped into a local minimum. As time proceeds, the temperature $T_k$ is decreasing and the

update tends to accept lower potential and the system is gradually running towards the global minimum. The original simulated annealing is very slow. In this paper, several novel techniques are used to accelerate the annealing procedure. They include the selection of a good initial motion field by using a multiresolution tree (MRT) based algorithm [13], the logarithmic rate temperature reduction, the stochastic relaxation of the control vector set $\mathbf{D_t}$ and a fast control vector search, etc. We refer to [12] for more details. By using all the acceleration techniques, we can get a BCV motion estimation algorithm which requires about twice of the computational cost of the exhaustive block search.

# 4 Coding of Displacement Frame Difference

As discussed in Section 2, the BCV-based displacement frame difference (DFD) is different from that of the conventional block based scheme. Inside each object, there is no block effect and the main distortion is geometrical deformation which is less sensitive subjectively. Around the boundaries of an object, the distortion is similar to that of the block based scheme. It is desirable to find a distortion measure that helps to select the DFD for encoding to enhance the overall performance of the encoder. We develop a criterion called the pixel threshold (PT) criterion which evaluates the DFD of each block by two factors: the position factor $p(a,b)$ and the error factor $e(a,b)$.

When the block is located at an object boundary, its position factor $p(a,b)$ takes value one; otherwise, it is zero. Thus, $p(a,b) = 1$ (or 0) means that block $B(a,b)$ is a boundary (or internal) block. The value $e(a,b)$ is the sum of the absolute value of DFD above a certain threshold $Q$ within a given block. Mathematically, it can written as

$$e(a,b) = \sum_{(x,y)\in B(a,b)} \left\lfloor \frac{|DFD(x,y,t)|}{Q} \right\rfloor, \tag{4.29}$$

where $Q$ is a threshold and $\lfloor x \rfloor$ means the largest integer value that is not greater than $x$. The PT decision is:

$$\begin{cases} p(a,b) = 0 \text{ and } e(a,b) > T_0 & \text{to be encoded} \\ p(a,b) = 1 \text{ and } e(a,b) > T_1 & \text{to be encoded} \\ \text{other cases} & \text{not to be encoded} \end{cases} \tag{4.30}$$

Note that in above $T_0$ and $T_1$ denote two different thresholds for internal and boundary blocks, respectively. It is often to choose $T_0 \gg T_1$, since the distortion of the internal block in the BCV scheme is mainly geometrical deformation which allows a larger value of error tolerance.

# 5 Experimental Results

Experiments are performed to compare the BCV scheme with the traditional block-based method used in the MPEG standard. The test video data include CLAIRE, MISSA, TREVOR, SALES, FOOTBALL and FLOWER sequences. The CLAIRE, MISSA, TREVOR and SALES sequences have a spatial resolution of $352 \times 288$ pixels and a frame rate of 10 frames per second. The FOOTBALL and FLOWER sequences have a spatial resolution of $352 \times 240$ pixels and a frame rate of 30 frames per second. We apply the exhaustive search to determine the motion vector for each block in the block-based method and use full-pel accuracy motion estimation for both the MPEG and BCV simulation. We do not implement the rate control for the video coder, instead, we set a constant MQUANT of 16. For the proposed BCV scheme, the control vectors and the displacement frame differences (DFD) are encoded by using the MPEG-I bit stream. Each boundary element is encoded by a predictive arithmetic coding scheme similar to the one in JBIG standard. We set the threshold for internal block quantization $T_0$ to be 3, the threshold for boundary block quantization $T_1$ to be 16. We split the DFD into macroblock and encode them according to the pixel threshold criterion as detailed in Section 4. It is worthwhile to point out that although the bit stream of BCV video coder is similar to that of the MPEG standard, they are essentially different in the motion field representation. Due to the incorporation of the boundary element, the BCV video coder/decoder is not compatible with the MPEG standard. However, it can be implemented by slightly modifying the MPEG code.

We show in Table 1 the average number of bits required to encode the motion field. Since the motion field in the BCV scheme consists of two components — the set $\mathbf{E_t}$ of boundary element functions and the set $\mathbf{D_t}$ of control vectors, we list the individual rate in the third and fourth rows in Table 1. The predictive gain listed in Table 1 is calculated as the average of the following instantaneous gain

$$\text{Gain}(t) = 10 \log \frac{255^2}{\sum DFD^2(x, y, t)}, \tag{5.1}$$

where the summation is over all pixels. We see from Table 1, the predictive gain of the BCV motion field is higher than the traditional full block matching method from 0.4 to 1.8 dB. This demonstrates that the BCV motion representation is superior to the block-based representation.

Experiments have also been performed to compare the BCV-based video coding scheme with the MPEG-I algorithm [6]. The results are shown in Table 2, where the rate is the average number of coding bits per pixel and the peak signal-to-noise ratio (PSNR) is calculated as the average of

| Item | CLAIRE | MISSA | SALES | TREVOR | FOOTBALL | FLOWER |
|---|---|---|---|---|---|---|
| Rate in bpp (full block match) | 0.0075 | 0.0108 | 0.0136 | 0.0186 | 0.0488 | 0.0320 |
| Rate in bpp (BCV) | 0.0063 | 0.0114 | 0.0101 | 0.0159 | 0.0418 | 0.0290 |
| Rate in bpp (BCV – $\mathbf{E_t}$) | 0.0002 | 0.0003 | 0.0003 | 0.0005 | 0.0003 | 0.0003 |
| Rate in bpp (BCV – $\mathbf{D_t}$) | 0.0061 | 0.0111 | 0.0098 | 0.0154 | 0.0415 | 0.0287 |
| Gain in dB (full block match) | 34.46 | 33.83 | 28.11 | 29.80 | 25.19 | 22.33 |
| Gain in dB (BCV) | 35.68 | 34.37 | 28.57 | 31.57 | 25.55 | 23.83 |

Table 1: Bit rates and gains for the coding of the motion field.

| Item | CLAIRE | MISSA | SALES | TREVOR | FOOTBALL | FLOWER |
|---|---|---|---|---|---|---|
| Rate in bpp (MPEG-I) | 0.0237 | 0.0251 | 0.0886 | 0.0819 | 0.2652 | 0.6028 |
| PSNR in dB (MPEG-I) | 35.31 | 34.49 | 28.77 | 31.04 | 29.67 | 26.15 |
| Rate in bpp (BCP) | 0.0080 | 0.0138 | 0.0679 | 0.0481 | 0.2196 | 0.4497 |
| PSNR in dB (BCP) | 35.81 | 34.62 | 28.86 | 31.77 | 29.92 | 26.84 |

Table 2: Bit rates and PSNR values for video sequence coding.

the instantaneous PSNR:

$$\mathrm{PSNR}(t) = 10 \log \frac{255^2}{\sum \left[ f(x,y,t) - \hat{f}(x,y,t) \right]^2}, \tag{5.2}$$

where the summation is over all pixels in an image. One can clearly see from Table 2 that, compared with MPEG-I, BCV video coding achieves a saving of 15-65% in the bit rate with nearly the same PSNR value. To further compare the performance of the MPEG-I and BCV, we plot the bit rate and the PSNR value as a function of the frame number for the CLAIRE and TREVOR test video sequences in Figs. 9 and 10, respectively.

Compared with the MPEG-I coded video, the BCV coded sequence appears to be smoother and clearer with less noisy patterns.

# 6 Conclusion and Extension

A new motion field representation and estimation framework based on the boundary-control vector (BCV) scheme and the Markov random field model was presented. The BCV motion field is generally continuous with discontinuity only at the object boundaries. It was demonstrated to be a promising method for video coding for the good rate-distortion performance. At present, the major

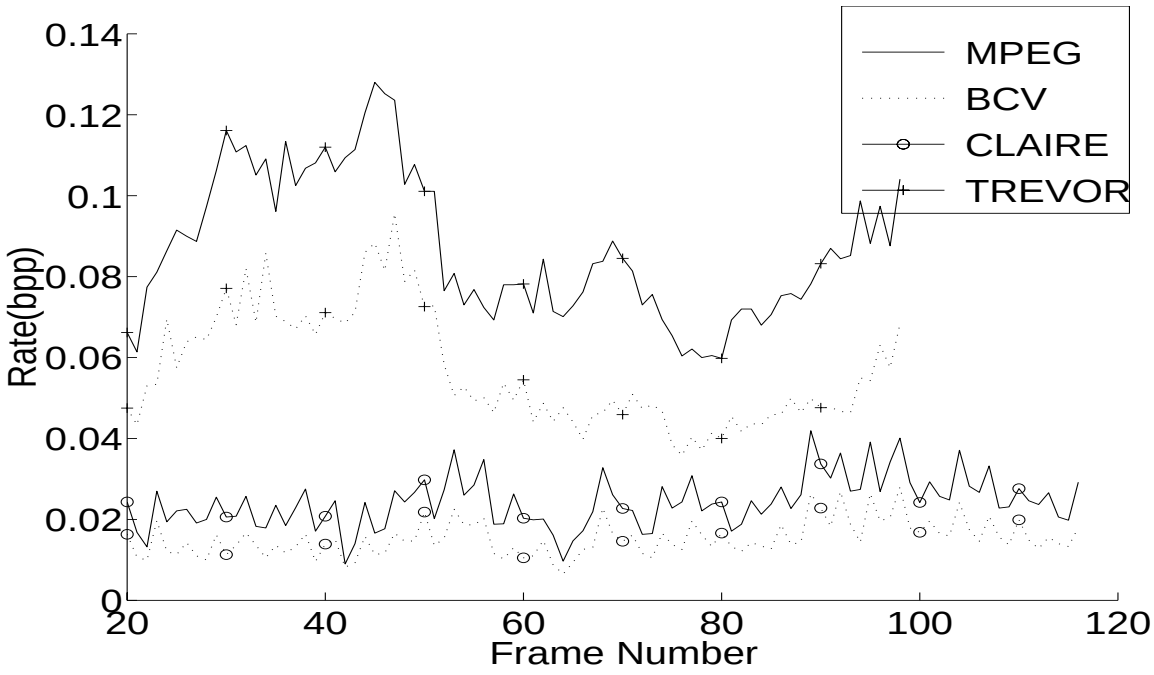



Figure 9: Comparison of coding rate between BCV scheme and MPEG-I scheme

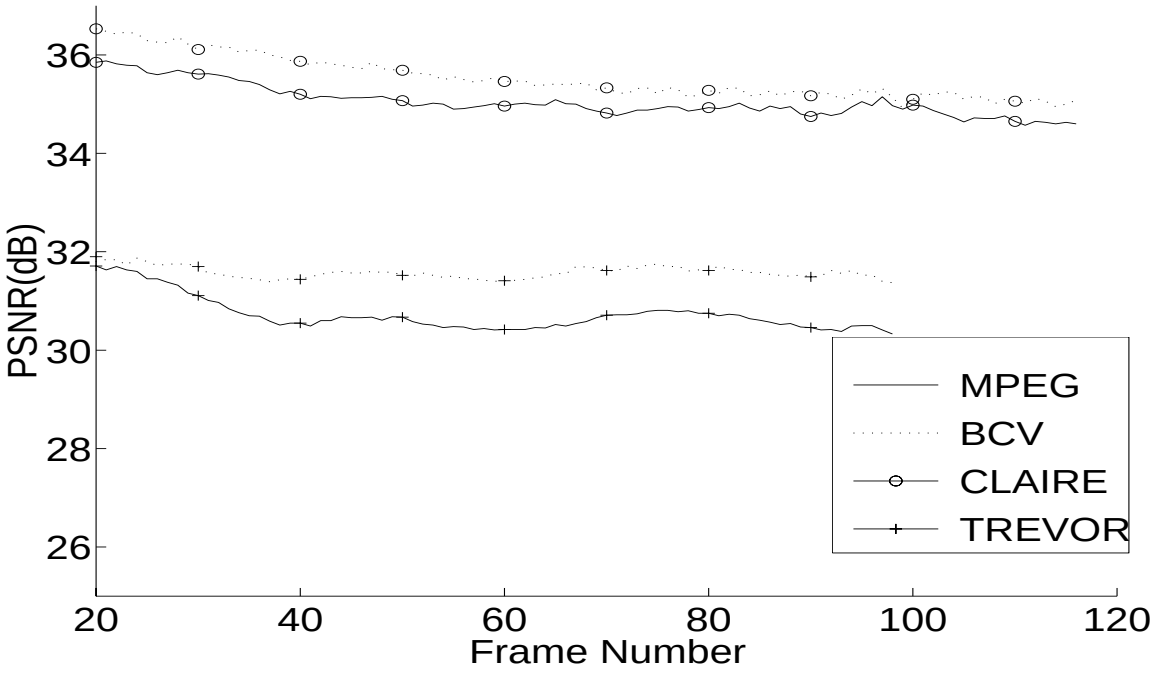



Figure 10: Comparison of coding PSNR between BCV scheme and MPEG-I scheme

disadvantage of the BCV scheme is its high computational complexity for motion filed estimation. We feel that the computational cost can be reduced by using a multiresolution approach, and a multiresolution BCV motion representation and estimation scheme is under our current study. Another interesting topic is to further improve the quality of the coded video images. We observe that even though the block effect in the DFD can be greatly reduced by using the BCV scheme, the subsequent DCT-based coding of the DFD still introduces the block effect. A better performance may be achieved by using a wavelet approach to encode the DFD.

# References


[1] C. Bergeron and E. Dubois, "Gradient based algorithms for block oriented MAP estimation of motion and application to motion compensated temporal interpolation," *IEEE Trans. on Circuits and Systems for Video Technology*, No. 1, pp. 72–85, Mar. 1991.

[2] N. Diehl, "Object-oriented motion estimation and segmentation in image sequences," *Image Communicaton*, No. 1, pp. 23–56, Feb. 1991.

[3] L. B. Feraud, M. Barlaud, and T. Gaidon, "Motion estimation involving discontinuities in a multiresolution scheme," *Optical Engineering*, No. 7, pp. 1475–1482, Jul. 1993.

[4] C. Fuh and P. Maragos, "Motion displacement estimation using affine model for image matching," *Optical Engineering*, No. 7, pp. 881–887, Jul. 1991.

[5] S. Geman and D. Geman, "Stochastic relaxation, gibbs distributions and the bayesian restoration of images," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, No. 6, pp. 721–741, Nov. 1984.

[6] ISO/IEC-JTC1/SC29/WG11, "Coding of moving pictures and associated audio for digital storage media at up to about 1.5 Mbit/s," ISO Standard CD 11172, ISO, 1991.

[7] T. Kimoto and Y. Yasuada, "Hierarchical representation of the motion of a walker and motion reconstruction for model-based image coding," *Optical Engineering*, No. 7, pp. 888–903, Jul. 1991.

[8] R. Kindeman and J. I. Snell, *Markov random fields and their applications*, Providence, RI: Amer. Math. Soc., 1990.

[9] K. Kirkpatrick, C. G. Jr., and M. Vechi, "Optimization by simulated annealing," *Science*, pp. 671–680, May. 1983.

[10] J. Konrad and E. Dubois, "Bayesian estimation of motion vectors," *IEEE Trans. on Pattern analysis and machine intelligence*, No. 9, Sep. 1992.

[11] H. J. Larson, *Introduction to probability theory and statistical inference*, John Wiley & Sons, 1974.

[12] J. Li, "High performance image compression: sequential and still," ph.d. dissertation, Tsinghua University, Tsinghua University, Beijing 100084, China, May. 1994.

[13] J. Li, X. Lin, and Y. Wu, "Multiresolution tree architecture with its application in video sequence coding," in *Visual Communication and Image Processing'93*, (Cambridge, MA), pp. 730–741, Nov. 8-11 , 1994.

[14] H. Musmann, M.Hotter, and J.Ostermann, "Object-oriented analysis-synthesis coding of moving images," *Image Communication*, No. 1, pp. 117–137, Feb. 1989.

[15] A. Netravali and J. Robbins, "Motion compensated television coding - Part I," *Bell Syst. Tech. J.*, pp. 631–670, Mar. 1979.

[16] C. C. of International Telegraph and Telephone, "video codec for audio-visual services at px64kb/s," SG XV Draft revision of recommendation H.261, CCITT, 1990.

[17] M. Orchard, "Predictive motion field segmentation for image sequence coding," *IEEE Trans. on Circuits and Systems for Video Technology*, No. 1, pp. 54–70, Jan. 1993.

[18] C. Papadopoulos and T. Clarkson, "Motion compensation using 2nd order geometric transformations," *Electronics Letters*, No. 25, pp. 2314–2315, Dec. 1992.

[19] L. L. Scharf, *Statistical signal processing: detection, estimation, and time series analysis*, Addison-Wesley Pub. Co., 1991.

[20] V. Seferidis, "Three dimensional block matching motion estimation," *Electronics Letters*, No. 18, pp. 1770–1771, Aug. 1992.

[21] V. Seferidis and M. Ghanbari, "General approach to block matching motion estimation," *Optical Engineering*, No. 7, pp. 1464–1474, Jul. 1993.

[22] P. Strobach, "Tree-structured scene adaptive coder," *IEEE Trans. on Communications*, No. 4, pp. 477–486, Apr. 1994.

[23] A. K. T. Aach and R. Mester, "Statistical model-based change detection in moving video," *Signal Processing*, Vol. 31, pp. 165–180, 1993.

[24] Y. Wang and O. Lee, "Active mesh — a video presentation scheme for feature seeking and tracking," in *Visual Communication and Image Processing'93*, (Cambridge, MA), Nov.8-11, 1993.

[25] F. Zhou, "A knowledge based coding of facial image and 34Mbit/s DPCM coding systems for composite PAL colour signal," dissertation, Ph.D. dissertation of Zhejiang University, Zhejiang, P.R.China, 1991.