Nonlinear Transform for Robust Dense Block-Based Motion Estimation

Rui Xu, David Taubman and Aous Thabit Naman

***Abstract*—We present a non-iterative multi-resolution motion estimation strategy, involving block-based comparisons in each detail band of a Laplacian pyramid. A novel matching score is developed and analysed in a spatially continuous setting. The proposed matching score is based on a class of non-linear transformations of Laplacian detail bands, yielding 1-bit or 2-bit representations that also have computational advantages. The matching score is evaluated in a dense full-search motion estimation setting, with synthetic content and optical flow datasets. Together with a strategy for combining the matching scores across resolutions, the proposed method is shown to produce smoother and more robust estimates than MSE in each detail band and combined. It tolerates more of non-translational motion such as rotation, validating the analysis, while providing much better localisation of the motion discontinuities. We also provide an efficient implementation of the motion estimation strategy and show that the computational complexity of the approach is comparable to the usual MSE block-based full-search motion estimation.**

***Index Terms*—non-linear transform, 1-bit and 2-bit representation, multi-resolution, block-based, motion estimation, dense field, full search.**

I. Introduction

Motion estimation, which is the process of finding pixel correspondences between a pair of video frames, is an important step in many visual signal processing applications. These include video communication applications, such as video compression and frame enhancement, as well as computer vision applications, such as segmentation, tracking and disparity estimation. A variety of approaches have been attempted to solve the problem. On one extreme we have block-based approaches, which aim to independently estimate the motion on image blocks; on the other extreme we have the optical flow approach, which involves a coupled optimisation problem to jointly optimise a local matching objective and a flow regularisation constraint. The aim of this paper is to propose and analyse a novel matching score other than mean square error (MSE), and evaluate its robustness in a dense full-search block-based true motion estimation setting. The objective is to minimise the motion error with respect to ground truth motion fields.

We confine our attention to translational dense block-based motion estimation. One motion vector is found for each pixel location by considering a block centred at each location; hence we refer to blocks over which the matching score is calculated as windows. This dense motion estimation is a good evaluation for block-based matching metrics such as the one proposed in this paper. However, within any given window, the motion in a real video frame is not generally translational. This non-translational motion particularly affects the contribution of high frequency content to the matching metric. It turns out that when MSE is used, these high frequency contributions can become useless or even misleading. Matching scores other than MSE have been proposed before such as MAD and Hadamard-MAD [1] but none of these address this issue explicitly.

In particular, we propose a multi-resolution matching score, which involves the decomposition of the original source frames into separate resolution detail subbands. A separate matching score for each resolution subband is designed to improve the robustness of motion estimation to the possible presence of non-translational motion within the windows. This is followed by the combination of matching scores generated from each resolution.

Within each resolution level, the matching score we propose involves a class of non-linear transformations of the content, producing intermediate thresholded and morphologically dilated versions of the detail band, which can be represented with one or two bits per sample. Some algorithms proposed in the literature also involve thresholded representations of frames. However, they usually focus on trading reduced matching quality for reduced matching complexity. Some are evaluated using the compensation error from the estimated motion field, e.g., [2], [3], or using the rate-distortion performance under a video compression scheme, e.g., [4]–[6]. Other such proposed matching scores are part of a larger application, e.g., [7]–[9]. [7] describes a 2-bit transform for increasing the robustness of motion estimation (registration) in a multi-frame enhancement application, from which we draw some of our inspiration. [9] proposes a local matching score in a probabilistic formulation to tell foreground from background in video frames. Most importantly, all of these methods except [9], calculate matching scores in the image domain. None of these methods involve the morphological dilation proposed in this work. None of these prior works provide an analysis to explain how these non-linear transforms can provide more robust motion estimation, nor do they use such an anlaysis to motivate the selection of parameters.

The use of multiresolution framework for motion estimation is certainly not new. However, for block-based methods they usually focus on the search strategy, using lower resolution to initialise the motion search of the next higher resolution – e.g., [10]–[12]. In fact, many optical flow methods involve this hierarchical approach as well – e.g., [15]–[17]. None of these prior works calculate a multi-resolution matching metric.

This work can be understood as an extension of the motion estimation strategy initially proposed in our earlier conference paper [18]. We build considerably on that work in following ways. We provide a much more comprehensive analysis of the properties of the non-linear transforms, which guides the selection of parameters in a manner that is sensitive to the expected potential for non-translational motion. We provide a much more comprehensive experimental evaluation, validating the analysis, while also explicitly showing that the proposed matching score resolves motion boundaries more effectively in the motion estimates despite the use of relatively large window size. We provide an efficient implementation of our approach and analyse its complexity, showing that the computational cost is closely related to that of traditional block-based motion estimation. Finally, we consider and evaluate other 1-bit and 2-bit transforms that can be understood within the same spatially continuous analysis.

We begin the paper in Section II by analysing the fundamental problem associated with MSE as a matching score, which serves as the motivation for the rest of the paper. Sections III, IV, V and VI describe the steps involved in the proposed approach, and provide analyses for each of these steps. Section VII is concerned with parameter selection based on the analyses. Section VIII gives a method for combining matching scores from all resolutions. Section IX further explains the search procedure and describes an efficient implementation with its computational complexity. Section X presents experimental results, firstly on a variety of synthetic sequences to test particular aspects of the proposed approach, and then on a standard optical flow dataset.

II. The effect of non-translational motion on MSE

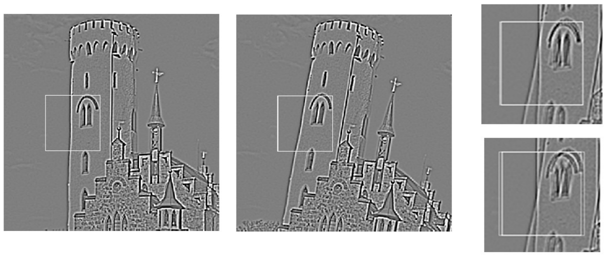


Fig. 1: Highest frequency detail bands $f^{(1)}$ and $g^{(1)}$, from two frames f and g, that are related by rotation and translation.

Consider two frames f and g. It is instructive to recognize that total squared error in the image domain is equivalent to the sum of the total squared errors found within each detail frequency subband. To see this, suppose we decompose f into a collection of narrow frequency bands $\{f^{(1)}, f^{(2)}, \ldots\}$ using ideal bandpass filters, such that $\hat{f}(\boldsymbol{\omega}) = \hat{f}^{(1)}(\boldsymbol{\omega}) + \hat{f}^{(2)}(\boldsymbol{\omega}) + \cdots + \hat{f}^{(k)}(\boldsymbol{\omega}) + \cdots$ and the regions of support $\Omega_k = \left\{\boldsymbol{\omega} \,\middle|\, \hat{f}^{(k)}(\boldsymbol{\omega}) \neq 0\right\}$ are disjoint. If we do the same for g then the total squared error associated with candidate motion vector $\boldsymbol{v}$ can be expressed as

$$\rho(\boldsymbol{v}) = \sum_k \frac{1}{(2\pi)^2} \iint_{\Omega_k} |\hat{g}^{(k)}(\boldsymbol{\omega}) - \hat{f}^{(k)}(\boldsymbol{\omega}) e^{-j\boldsymbol{\omega}^t \boldsymbol{v}}|^2 d\boldsymbol{\omega}.$$

If f and g are related by a translational displacement $\boldsymbol{\sigma}_0$, so that $g[\boldsymbol{n}] = f_{\boldsymbol{\sigma}_0}[\boldsymbol{n}]$ (i.e., $\hat{g}(\boldsymbol{\omega}) = \hat{f}(\boldsymbol{\omega}) e^{-j\boldsymbol{\omega}^t \boldsymbol{\sigma}_0}$), then $\rho(\boldsymbol{v})$ takes its minimum value of 0 when $\boldsymbol{v} = \boldsymbol{\sigma}_0$. If $\boldsymbol{v}$ differs from $\boldsymbol{\sigma}_0$ by some small motion estimation error $\boldsymbol{\delta}$, we have

$$\rho(\boldsymbol{v}) = \sum_k \underbrace{\frac{1}{(2\pi)^2} \iint_{\Omega_k} |\hat{g}^{(k)}(\boldsymbol{\omega})|^2 \cdot \underbrace{|1 - e^{-j\boldsymbol{\omega}^t \boldsymbol{\delta}}|^2}_{\approx |\boldsymbol{\omega}^t \boldsymbol{\delta}|^2} d\boldsymbol{\omega}}_{\rho^{(k)}(\boldsymbol{v})} .$$

Evidently, the effect of small motion estimation errors is greatest at the highest spatial frequencies. In practice, we do not expect two frames to be related by pure translational motion, even over a limited window. This means that no matter what motion vector $\boldsymbol{v}$ is selected, there are always displacement errors and these errors most strongly affect the higher spatial frequency components, represented by bands $f^{(k)}$ for which k is small. Accordingly, the contributions $\rho^{(k)}(\boldsymbol{v})$ from these bands to the overall squared error $\rho(\boldsymbol{v})$ are useless at best and misleading at worst.

This phenomenon can readily be visualised in the spatial domain, by considering the highest frequency bands $f^{(1)}$ and $g^{(1)}$ of two frames f and g that are related by a combination of translation and rotation. Within a sufficiently small spatial neighbourhood, rotation can always be approximated by translation; however, the window size employed for motion estimation must be large enough to ensure that the motion is likely to be observable. Figure 1 illustrates the matching problem in this high frequency band. Clearly, no matter what motion vector $\boldsymbol{v}$ is selected, motion estimation errors affect most of the window and the matching score is very poor. In the example, many vectors $\boldsymbol{v}$, including the true motion, result in most of the large values of $g^{(1)}$ overlapping values that are close to 0 within $f_{\boldsymbol{v}}^{(1)}$, so that the total squared error is both large and highly insensitive to $\boldsymbol{v}$. As a result, we expect the $\rho^{(1)}(\boldsymbol{v})$ component of $\rho(\boldsymbol{v})$ to be unhelpful or even misleading.

The above observation represents the primary inspiration for the work described in this paper. Since the high frequency components play an important role in localising image features such as edges, our goal is to devise matching criteria that enhance their utility, despite the presence of non-translational motion. It is tempting to apply a low-pass filter to these subbands in the hope of "spreading" the image features prior to matching. However, when the bands are sufficiently narrow, applying any LSI filter to $f^{(k)}$ and $g^{(k)}$ prior to evaluating the squared error has no effect other than to scale $\rho^{(k)}$ by a constant; this does affect the relative contribution of each component $\rho^{(k)}(\boldsymbol{v})$ to the overall matching score, but has no impact on the robustness of the individual scores $\rho^{(k)}(\boldsymbol{v})$. This suggests that a nonlinear transformation is required to fully exploit the information available within the individual subbands. In Sections III, IV, V and VI, we propose such a way to improve the reliability of the matching scores associated with higher frequency bands.

III. Multiresolution approach

A. Proposed Laplacian Pyramid

In view of the roles played by different spatial frequency components, as discussed in Section II, we choose to decompose each frame into frequency bands $\{f^k\}$ and $\{g^k\}$, evaluating a matching score for each candidate motion vector $\boldsymbol{v}$ separately within each subband, after which the scores can be combined. The specific transform employed in this work is shown in Figure 2, where G_σ is a Gaussian filter with $\sigma = \sqrt{3}$ and $f[\boldsymbol{n}]$ is first interpolated by 2 to get $f_{\times 2}$. The choice $\sigma = \sqrt{3}$ ensures that each detail band is essentially free from aliasing artefacts, as we shall see shortly.

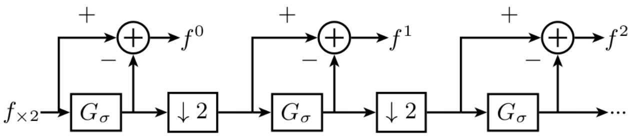


Fig. 2: Laplacian Pyramid for Analysis

B. Modelling Subbands of a Continuous Ideal Edge

To investigate matching MSE and the proposed non-linear matching in subbands, we will find it useful to model an ideal edge $u(x)$ in the continuous domain in one dimension using the Heaviside step function $u(x) = H(x) = \int_{-\infty}^{x} \delta(\tau) d\tau$, where $\delta(x)$ is the Dirac delta function. This ideal edge is subjected to a low pass Gaussian $G_{\sigma_I}(x) = \frac{1}{\sigma_I \sqrt{2\pi}} \exp(-\frac{x^2}{2\sigma_I^2})$ to simulate the effect of a bandlimited imaging system, yielding the imaged edge

$$u_I(x) = (u * G_{\sigma_I})(x) = \frac{1}{2}[\text{erf}(\frac{x}{\sqrt{2}\sigma_I}) + 1],$$

where $\text{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x \exp(-\tau^2) d\tau$ is the error function that has $\text{erf}(\infty) = 1$ and $\text{erf}(-\infty) = -1$.

Let $u^k(x)$ be the Laplacian subband produced by such an edge $u_I(x)$ at levels $k = 0, 1, 2, 3...$ with unit sampling rate such that $u^k[n] = u^k(x) \mid_{x=n}$. Following the proposed Laplacian pyramid in Figure 2, having $f_{\times 2} = u_I$, u^k can be calculated in the continuous domain using the following equations.

$$\begin{aligned} u^0 =& u * G_{\sigma_I} - u * G_{\sigma_I} * G_\sigma, \\ u^1 =& S_2(u * G_{\sigma_I} * G_\sigma) - [S_2(u * G_{\sigma_I} * G_\sigma)] * G_\sigma \\ =& u * G_{\frac{\sigma_I}{2}} * G_{\frac{\sigma}{2}} - u * G_{\frac{\sigma_I}{2}} * G_{\frac{\sigma}{2}} * G_\sigma, \\ & ... \\ u^k =& u * G_{\frac{\sigma_{(k-1)}}{2}} - u * G_{\sigma_k}, \end{aligned}$$

where S_2 is the sub-sampling operator in the continuous domain such that $(S_2 \circ f)(x) = f(2x)$, and $\sigma_k^2 = \sigma_I^2 2^{-2k} + \sigma^2 2^{-2k} + ... + \sigma^2 2^{-2\times 1} + \sigma^2 2^{-2\times 0}$, for $k \geq 0$. The above series quickly converges to $\sigma_k^2 = \frac{4}{3}\sigma^2$, and hence $u^k(x)$ quickly converges to $U(x)$, which we define as the universal subband edge function,

$$U(x) = \frac{1}{2}[\text{erf}(\frac{x}{\sqrt{2}\sigma/\sqrt{3}}) - \text{erf}(\frac{x}{2\sqrt{2}\sigma/\sqrt{3}})]. \qquad (1)$$

In fact, for the specific choice of $\sigma = \sqrt{3}$, when $\sigma_I = 1$, $u^k(x) = U(x)$ for all $k \geq 0$. The choice of $\sigma = \sqrt{3}$ and the reasonable assumption of $\sigma_I = 1$ results in the effective standard deviation $\sigma_k = 2$ for all $k \geq 0$, which guarantees that the signal before sub-sampling is bandlimited to $\omega \in [-\frac{\pi}{2}, \frac{\pi}{2}]$, so that the continuous analysis here is not invalidated by aliasing from the discrete implementation.

In two dimensions, the subband edge function derived above describes the horizontal component of a vertical edge, as it appears in each resolution level. Using rotationally invariant Gaussian filters, the same model applies to edges of any orientation, describing their behaviour as a function of displacement from the edge.

C. Summary of the Approach

At this point, we provide a summary of the overall implementation. In the ensuing sections, we consider the individual steps, providing an accompanying analysis of each step. These steps involve thresholding, dilation, matching, combining matching scores across resolutions. Figure 3 gives this summary as a flow chart.

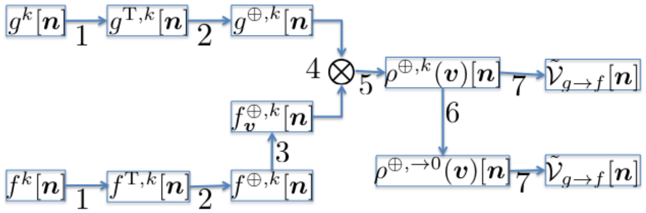

$g^k[\boldsymbol{n}]$
1
$g^{\mathrm{T},k}[\boldsymbol{n}]$
2
$g^{\oplus,k}[\boldsymbol{n}]$
4
5
$\rho^{\oplus,k}(\boldsymbol{v})[\boldsymbol{n}]$
7
$\tilde{\mathcal{V}}_{g\to f}[\boldsymbol{n}]$
6
$f_{\boldsymbol{v}}^{\oplus,k}[\boldsymbol{n}]$
3
$\rho^{\oplus,\to 0}(\boldsymbol{v})[\boldsymbol{n}]$
7
$\tilde{\mathcal{V}}_{g\to f}[\boldsymbol{n}]$
$f^k[\boldsymbol{n}]$
1
$f^{\mathrm{T},k}[\boldsymbol{n}]$
2
$f^{\oplus,k}[\boldsymbol{n}]$

Fig. 3: This flow chart shows the multi-resolution approach primarily on a single resolution level k. $g^k[\boldsymbol{n}]$ and $f^k[\boldsymbol{n}]$ are discrete representations of subbands from Figure 2. The other symbols are defined properly in their respective sections. The labelled operations 1, 2 and etc in the figure are: 1.Thresholding (Section IV) 2.Dilation (Section V) 3.Whole frame shift (refer to Section IX on computational complexity) 4.Pixelwise XOR operation (Section VI) 5.Moving average counter (Section VI) 6.Weighted combination across all resolutions (Section VIII) 7.The motion estimate $\tilde{\mathcal{V}}_{g\to f}[\boldsymbol{n}]$ is obtained by choosing the candidate motion vector $\boldsymbol{v}$ that optimizes the matching score at each pixel location $\boldsymbol{n}$ (Section IX)

IV. Thresholding

In this section we introduce a set of non-linear transformations of the subband data, which allow edge features in the high frequency subbands to be effectively expanded, avoiding the problem introduced in Section II.

The simplest non-linear transformation of interest maps the Laplacian subbands $g^k[\boldsymbol{n}]$ to binary-valued representations $g^{\mathrm{B},k}[\boldsymbol{n}]$, according to

$$g^{\mathrm{B},k}[\boldsymbol{n}] = \begin{cases} 1 & |g^k[\boldsymbol{n}]| > (|g^k| * h_L)[\boldsymbol{n}], \\ 0 & \text{otherwise}, \end{cases} \qquad (2)$$

where the absolute value of g^k is first subject to a low-pass spreading filter h_L, which is then used to threshold the absolute value of g^k.

This thresholding operation can be understood as a contrast normalization step, where $g^k[\boldsymbol{n}]$ values are divided by the average absolute values of their neighbours, after which the absolute value of the result is subjected to a threshold of 1. Contrast normalization without thresholding is much more problematic, being an ill-conditioned operation, requiring the use of explicit division operators. Additionally, the binary representation is more amenable to efficient matching and morphological processing operations.

One improvement on this theme is to keep the signs by using ternary-valued representations so that the two sides of an edge can be distinguished,

$$g^{\mathrm{T},k}[\boldsymbol{n}] = \begin{cases} 1 & g^k[\boldsymbol{n}] > +(|g^k| * h_L)[\boldsymbol{n}], \\ -1 & g^k[\boldsymbol{n}] < -(|g^k| * h_L)[\boldsymbol{n}], \\ 0 & \text{otherwise.} \end{cases} \tag{3}$$

We refer to this as the 2-bit-abs method. Yet a further improvement is to obtain $g^{\mathrm{T},k}$ by developing positive and negative thresholds separately,

$$g^{\mathrm{T},k}[\boldsymbol{n}] = \begin{cases} 1 & g^k[\boldsymbol{n}] > (g^{k+} * h_L)[\boldsymbol{n}], \\ -1 & g^k[\boldsymbol{n}] < (g^{k-} * h_L)[\boldsymbol{n}], \\ 0 & \text{otherwise,} \end{cases} \tag{4}$$

where $g^{k+} = \max(0, g^k)$ and $g^{k-} = \min(0, g^k)$ are the +ve and -ve parts of g^k. This variation turns out to be better adapted to the representation of edges for which the neighbouring positive and negative lobes in the Laplacian detail band are not of equal magnitude – this may happen when two ideal edges lie close together. We refer to this as the 2-bit method.

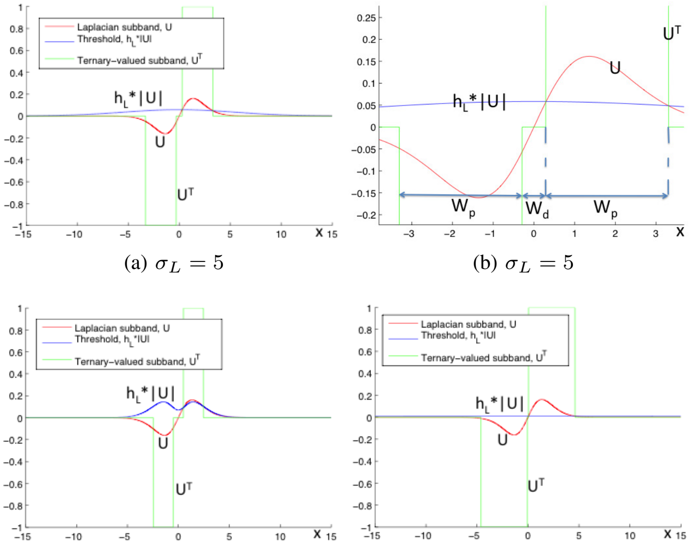
In addition, we can add an extra noise threshold δ in the above Equations (2), (3) and (4), such that the subband values have to be also larger than δ or smaller than $-\delta$ for the transformed values to be non-zero. This seems like a reasonable thing to do to avoid excessive contribution from noise. In fact, in a related work [3], this is quite useful. However, it turns out that our proposed approach for combining matching scores renders such noise threshold useless, as we show in our experiments.

In the next section, we analyze the effect of these thresholding operations in order to understand the impact of the selected spreading filter h_L on the geometry of the ternary and binary subband images; this analysis drives the selection of h_L in the rest of the paper.

A. Continuous Domain Analysis of Thresholding

Figure 4a illustrates the impact of thresholding on our universal subband edge function $U(x)$ defined in Equation (1). The key features of interest are the widths and separation of the non-zero regions (or "peaks") in the thresholded output, that arise on either side of the underlying edge feature; we denote these features W_p and W_d. For our analysis, we take the low pass filter h_L to be AG_{σ_L}, where G_{σ_L} is a normalized Gaussian filter and A is the DC gain.

We begin by considering the 1-bit and 2-bit-abs methods, for both of which the thresholding function is obtained by convolving h_L with $|U(x)|$. We then extend the approach to include the full 2-bit method, for which separate positive and negative threshold functions are obtained by convolving h_L with $U^+(x)$ and $U^-(x)$. Although $U(x)$ is Nyquist bandlimited, its absolute value and its positive and negative parts are not. For this reason, the continuous domain analysis with which we begin may be inaccurate. We assess the level of accuracy later, by considering the impact of aliasing.



(a) $\sigma_L = 5$

(b) $\sigma_L = 5$

(c) $\sigma_L = 0.5$

(d) $\sigma_L = 30$

Fig. 4: This figure shows the 2-bit transform for $\sigma_L = 5$ and some extreme σ_L values for $U(x)$ with gain $A = 1$ and noise threshold $\delta = 0$. (b) zooms in on (a) to show W_p and W_d

1) Continuous domain modelling of the 1-bit and 2-bit-abs methods: In both the 1-bit and 2-bit-abs methods, in Equations (2) and (3), $g^{\mathrm{B},k}[\boldsymbol{n}]$ and $g^{\mathrm{T},k}[\boldsymbol{n}]$ are non-zero if and only if $|g^k[\boldsymbol{n}]| - (h_L * |g^k|)[\boldsymbol{n}] > 0$. Adopting our uniform subband edge function, and restricting the analysis to one dimension, $|g^k[n]| = |U(x)|_{x=n}$. If we temporarily ignore the fact that $|U(x)|$ cannot be Nyquist bandlimited, the discrete convolution of the sampled Gaussian $h_L[n]$ with $g^k[n]$ can be seen as equivalent to sampling the result produced by the continuous convolution of $h_L(x) = AG_{\sigma_L}(x)$ with $|U(x)|$ – this is because $G_{\sigma_L}(x)$ itself is effectively Nyquist bandlimited for any $\sigma_L \gtrsim 1$. This allows us to find the W_p and W_d parameters by considering the width and separation of positive values of the continuous function $(|U| - A \cdot (G_{\sigma_L} * |U|))(x)$. Depending on the precise location of the edge, with respect to the sampling grid, the discrete thresholded output may exhibit widths of either $\lfloor W_p \rfloor$ or $\lceil W_p \rceil$ and separations of either $\lfloor W_d \rfloor$ or $\lceil W_d \rceil$.

We evaluate the function

$$\begin{aligned} p(x) &= (|U| - A \cdot (G_{\sigma_L} * |U|))(x) \\ &= |U(x)| - \frac{A}{\sqrt{2\pi\sigma_L^2}} \int e^{-\tau^2/(2\sigma_L^2)} |U(x-\tau)| d\tau \end{aligned}$$

via numerical integration and report the width and separation of the regions for which $p(x) > 0$, in Figure 5, for various combinations of the parameters A and σ_L. Figure 4 plots the threshold function (in blue) and the 2-bit-abs thresholded output $U^T(x)$ (in green), where $U^T(x)$ equals sign$(U(x))$ if $p(x) > 0$ and 0 otherwise. The 1-bit method produces $U^B(x) = |U^T(x)|$.

2) Continuous domain modelling of the 2-bit method: In the 2-bit method defined in Equation (4), $g^{\mathrm{T},k}[\boldsymbol{n}] = 1$ if and only if $|g^{k+}|[\boldsymbol{n}] - (h_L * |g^{k+}|)[\boldsymbol{n}] > 0$. Adopting our universal subband edge function, restricting the analysis to

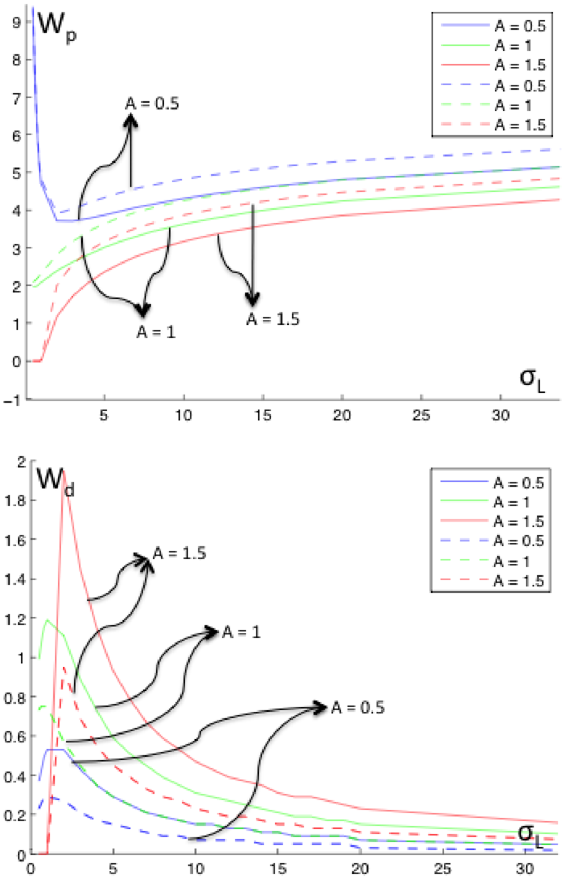


Fig. 5: W_p and W_d as a function of σ_L. The solid line is using $|U|$ and the dashed line is using $U^{\pm}$

.

one dimension, and temporarily ignoring the fact that $U^{+}(x)$ cannot truly be Nyquist bandlimited, we proceed as before, deducing the width of the positive region as that for which $p^{+}(x) > 0$, where

$$\begin{aligned} p^{+}(x) &= (|U^{+}| - A \cdot (G_{\sigma_L} * |U^{+}|))(x) \\ &= |U^{+}(x)| - \frac{A}{\sqrt{2\pi\sigma_L^2}} \int e^{-\tau^2/(2\sigma_L^2)} |U^{\pm}(x-\tau)| d\tau \end{aligned}$$

The width of the negative region, corresponding to $|g^{k-}|[\boldsymbol{n}] - (h_L * |g^{k-}|)[\boldsymbol{n}] > 0$, is exactly the same in this continuous domain analysis, due to the symmetry of $U(x)$. The separation between the positive and negative regions is also readily found, as twice the $\min\,(x > 0 \mid p^{+}(x) = 1)$. The W_p and W_d values produced in this way are shown as dashed lines in Figure 5.

Evidently, for a given value of σ_L, the 2-bit method requires A to be about twice as large as the 1-bit and 2-bit-abs methods, in order for W_p and W_d to be comparable (see the dashed green and solid blue curves in Figure 5). This observation holds for modest to large values of σ_L, which makes sense considering that the mean value of $U^{+}(x)$ and of $U^{-}(x)$ are each only half that of $|U(x)|$. As a result, the behaviour of all three thresholding methods is essentially the same, subject to appropriate choice of A, except where σ_L is very small. The only significant differences between the methods are that the 2-bit and 2-bit-abs methods preserve the sign information that distinguishes light-dark from dark-light transitions, while the 2-bit method is expected to handle multiple nearby edges more reliably than the 2-bit-abs method.

3) Impact of aliasing on the analysis: We now turn briefly to study the aliasing of the non-linear operations on $U(x)$ to show that the above continuous domain analysis is not invalidated by the discrete implementation. The discrete implementation yields $(h_L * |g^k|)[\boldsymbol{n}]$, which is equivalent to sampling the continuous function $(h_L * |g^k|)(\boldsymbol{x})$, so long as the aliasing contributions from $\widehat{|g^k|}(\omega)$ at $|\omega| > \pi$ are negligible. Since h_L is a narrow-band filter, selecting only frequencies close to DC, it is sufficient to consider only the aliasing contributions that map to DC; these correspond to $\widehat{|g^k|}(2m\pi)$ for non-zero integers m. More particularly, the relative impact of aliasing on the continuous domain analysis for the universal subband edge function $U(x)$ is governed by the ratio $\left|\frac{\widehat{|U|}(2m\pi)}{\widehat{|U|}(0)}\right|$. Evaluating this ratio, we find that

$$\left|\frac{\widehat{|U|}(2\pi)}{\widehat{|U|}(0)}\right| = 0.0133, \quad \left|\frac{\widehat{|U|}(4\pi)}{\widehat{|U|}(0)}\right| = 0.0032.$$

Similarly, we can find $\widehat{U^{\pm}}(\omega)$. Although, the shape of $\widehat{U^{\pm}}(\omega)$ is different from $\widehat{|U|}(\omega)$, the ratios $\left|\frac{\widehat{U^{\pm}}(2m\pi)}{\widehat{U^{\pm}}(0)}\right|$ turn out to have exactly the same values. We conclude that our continuous domain analysis provides a reliable indication of the ternary/binary image feature widths and separations produced by the actual discrete implementation.

V. Morphological Dilation

To increase tolerance to small motion errors during the matching process, $g^{\mathrm{T},k}$ is further processed by morphological dilation. Specifically, we form $g^{\oplus,k}$ by two successive applications of the operator $\oplus_{\overline{11}}$, where

$$\left(\oplus_{\overline{11}} x\right)[\mathbf{n}] = \bigvee_{\mathbf{k}\in B_\epsilon, \ni x[\mathbf{n}-\mathbf{k}]\neq 11} x[\mathbf{n}-\mathbf{k}]. \qquad (5)$$

Here, B_ϵ is the circular structuring set with radius ϵ, and $\bigvee$ denotes the binary inclusive OR operation, applied to each bit-plane of the 2-bit ternary-valued subband,

$$x[\mathbf{n}] = \begin{cases} 01 & g^{\mathrm{T},k} = 1, \\ 10 & g^{\mathrm{T},k} = -1, \\ 00 & \text{otherwise}. \end{cases}$$

Although the initial ternary-valued subband cannot take the value 11, after dilation such values do generally occur; however, the operator $\oplus_{\overline{11}}$ dilates only those locations that are either strictly +ve (01) or strictly -ve (10), not both.

A. Dilation in the Continuous Domain

Figure 6 illustrates the dilation process in the continuous domain. The overall dilation is a two-step process, where the first step increases the separation between the 01 and 10 regions, i.e. $W_d^1 > W_d$, and the second step makes the two bands wider. Specifically, after the first dilation,

$$\begin{aligned} W_d^1 &= 2\epsilon - W_d, \\ W_p^1 &= W_p + W_d, \end{aligned}$$

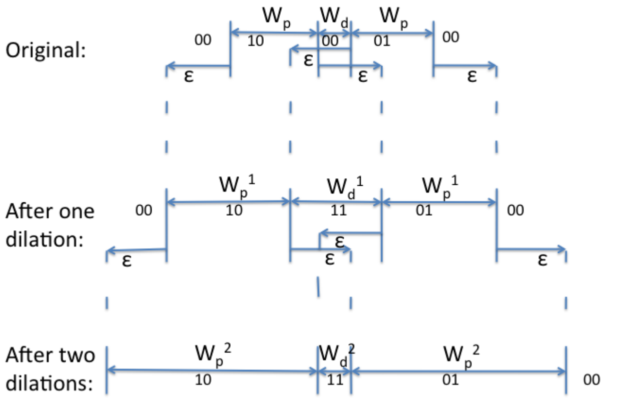


Fig. 6: Illustration of Dilation in the Continuous Domain

and after the second dilation,

$$W_d^2 = W_d,$$
$$W_p^2 = W_p + 2\epsilon.$$

In real applications, the width of band W_p^2, which is used for matching, also depends on the density of edges. The presence of a second nearby edge can limit the dilation of a first edge. The fact that $\oplus_{\bar{1}1}$ does not dilate locations in which the positive and negative bit-planes overlap is particularly advantageous in such situations, since it prevents dense edges features from merging into each other.

Where the 1-bit method is used to form binary valued subbands, instead of the ternary valued subbands considered above, we simply dilate once by the circular structuring element B_ϵ. This produces a single non-zero band of width $W_p^2 = (2W_p + W_d) + 2\epsilon$ around an edge, so long as $2\epsilon \geq W_d$.

VI. Matching on 2-bit transforms

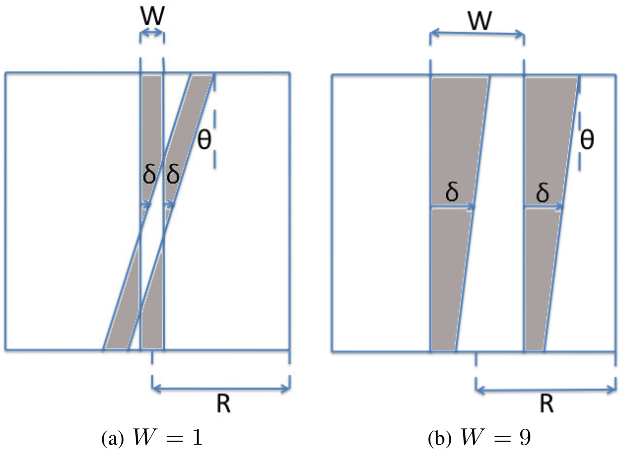
A matching score can now be calculated between $g^{\oplus,k}[\boldsymbol{n}]$ and $f_{\boldsymbol{v}}^{\oplus,k}[\boldsymbol{n}]$. We do this by adding the MSE's between corresponding bit-planes of $g^{\oplus,k}$ and $f_{\boldsymbol{v}}^{\oplus,k}$, evaluated over a window $\mathsf{W}_{\boldsymbol{n}}$ of size $(2R+1)^2$ centered at location $\boldsymbol{n}$, where R is the half-width of the square matching window. This matching score can be evaluated readily by taking the exclusive OR between the corresponding bit-planes and counting the number of 1s in the result. Specifically, we have

$$\rho^{\oplus,k}(\boldsymbol{v})[\boldsymbol{n}] = \sum_{\boldsymbol{p}\in\mathsf{W}_{\boldsymbol{n}}} \left[(g^{\oplus,k}[\boldsymbol{p}])_{\text{bit0}} \text{XOR} (f_{\boldsymbol{v}}^{\oplus,k}[\boldsymbol{p}])_{\text{bit0}}\right] + \left[(g^{\oplus,k}[\boldsymbol{p}])_{\text{bit1}} \text{XOR} (f_{\boldsymbol{v}}^{\oplus,k}[\boldsymbol{p}])_{\text{bit1}}\right]. \tag{6}$$

Naturally, where the 1-bit method has been used, the dilated images are represented by only one bit-plane and the exclusive OR operation is applied only on this bit-plane.

A. Matching Two Peak Bands in the Continuous Domain

Figure 7 helps to understand the impact of motion estimation errors $\boldsymbol{\delta}$ on the matching score, considering only one of the bit-planes of the proposed 2-bit transform [1]. The figure is equally helpful in understanding the impact of motion errors on the single bit-plane produced by the 1-bit method. In each case, the edge feature produces a band of non-zero values of width $W \triangleq W_p^2$, as explained above, and we consider motion errors $\boldsymbol{\delta}$ in the normal direction to the edge. Importantly, the matching score is affected by relative warping of the edge between the two frames, in addition to translation. For simplicity, we model this warping as a local rotation through angle θ; as a result, the matching score is not 0, even when the translational error $\boldsymbol{\delta}$ at the centre of the window is $\mathbf{0}$. This renders the matching score more susceptible to the impact of noise. To see this, we begin by noting that the matching score by the mismatched edge features is simply the area of the shaded region in Figure 7. We can model this matching score carefully in the continuous domain, as a function of the motion error $|\boldsymbol{\delta}|$; the result is shown in Figure 8. Note that this result is independent of the resolution level, assuming that matching is performed on a window of size $(2R+1)\times(2R+1)$, measured relative to the sample spacing in that resolution level.

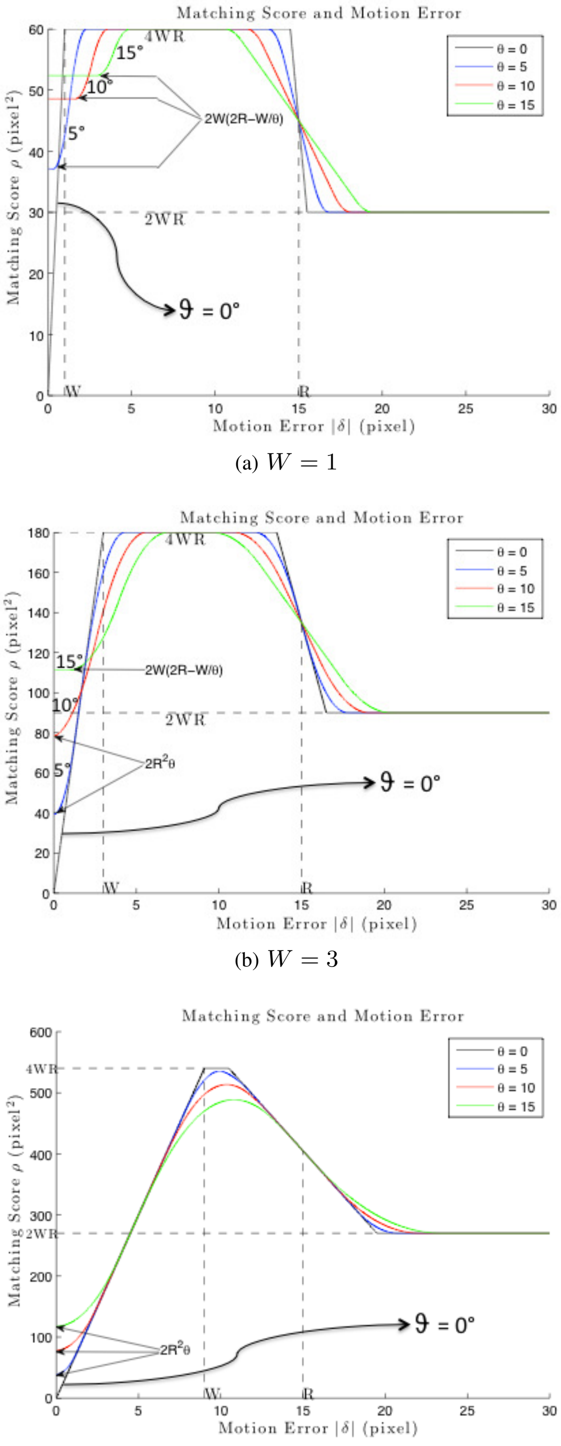


(a) $W = 1$ (b) $W = 9$

Fig. 7: Interpretation of the matching score, where $R = 15$ is the half-width of the square matching window.

In general, for small θ and assuming that $W > R\theta$, the minimum value of the matching score always corresponds to the correct matching location at $|\boldsymbol{\delta}| = 0$, no matter how large the search range is. In fact, under these conditions, the difference in the matching scores produced by large motion errors $|\boldsymbol{\delta}|$ and the ideal case where $\boldsymbol{\delta} = \mathbf{0}$ is given by $2R(W - R\theta)$. What this means is that the larger the width W of the edge feature, the less likely it is that large motion estimation errors result from the combination of noise and geometric warping (rotation here). Figure 8 reveals that there is a "capture region" around the ideal location at $\boldsymbol{\delta} = \mathbf{0}$, where the matching score is highly sensitive to the motion vector $\boldsymbol{v}$; beyond this point, the matching score becomes insensitive to $\boldsymbol{\delta}$, except where $|\boldsymbol{\delta}|$ is so large that the edge feature is found within the window in only one of the two frames. The size of the capture region can also be taken as a limit on the coarsest step size that may be used during a reliable motion search. Evidently, the capture region is smallest in the absence of rotation, where its extent is given by W. Together, these two observations explain the role of

[1] For isolated edges, the contribution of each bit-plane to the matching score in equation (6) is exactly the same from the continuous domain perspective.



(a) $W = 1$

(b) $W = 3$

(c) $W = 9$

Fig. 8: Matching score ρ a function of motion error $|\boldsymbol{\delta}|$, where $R = 15$ is the half-width of the square matching window.

the dilation step, which increases W and hence improves the reliability of the motion search subject to relative geometric warping and/or limited precision in the search step. Recall that the MSE matching criterion on original subband images is highly sensitive to motion errors induced by non-translational motion and/or limited search precision.

VII. Parameter Selection

The above analysis provides some insight into the selection of parameters for our proposed non-linear transform and matching process. Specifically, the parameters of interest are the dilation radius ϵ, the spreading filter gain and extent parameters A and σ_L, and the matching window size R. The first three parameters determine the widths W of the edge features produced in the binary or ternary valued subbands, after dilation. The relationship between W and R then determines the robustness of the matching process.

To obtain one possible choice of parameters to use in our experiments, we pick the value for R first. R has to be large enough to include sufficient features for motion to be observable within the matching window. For this reason, we pick $R = 14$, which is a 29×29 window for all subbands. For the highest resolution subband, this is equivalent to a 15×15 window in the original frame resolution. Given the amount of warping that we want to tolerate in terms of rotation θ, we can then pick W, such that $W \geq 2R\theta$. Arbitrarily selecting a geometric tolerance of $|\theta| \leq 18°$, we obtain $W = 9$. We choose to implement h_L through a cascade of two 13×13 moving average window filters, with unit DC gain; this crudely approximates a Gaussian filter with $\sigma_L \approx 5$ and $A = 1$. From Figure 5, this choice yields $W_p \approx 3$, so that the discrete implementation produces edge features whose widths lie in the range 2 to 4 prior to dilation. The final target width of $W_p^2 \approx 9$ is achieved by selecting $\epsilon = 3$ for the dilation parameter.

To summarise, we pick the following parameters in the evaluation and application Sections X and X-C, unless stated otherwise: h_L is implemented by applying a 13×13 moving average filter twice, $\epsilon = 3$ and $R = 14$.

VIII. Combining multiresolution matching scores

We turn now to the problem of combining the matching scores generated at each level of the Laplacian pyramid. For a given candidate displacement $\boldsymbol{v}$, we generate matching scores $\rho^{\oplus,k}(\boldsymbol{v})[\boldsymbol{n}]$ at each location $\boldsymbol{n}$ of each level k, using (6) – note that this is nothing other than a moving average, which can be implemented recursively with extremely low complexity, independent of the window size $(2R+1)^2$. We then progressively interpolate and combine the matching scores from coarser levels into finer levels according to

$$\rho^{\oplus,\to k}(\boldsymbol{v})[\boldsymbol{n}] = \rho^{\oplus,k}(\boldsymbol{v})[\boldsymbol{n}] + (1 - c^{\mathrm{M},k}[\boldsymbol{n}]) \cdot \bar{\rho}^{\oplus,\to k+1}(\boldsymbol{v})[\boldsymbol{n}] \tag{7}$$

where $\bar{\rho}^{\oplus,\to k+1}(\boldsymbol{v})$ is the interpolated version of $\rho^{\oplus,\to k+1}(\boldsymbol{v})$ and $c^{\mathrm{M},k}[\boldsymbol{n}]$ is an estimate of the reliability of the level k estimates. In this paper, we base this reliability estimate on a recently introduced measure of the local image structure (as opposed to unstructured noise).

Specifically, $c^{\mathrm{M},k}$ is the structural content estimator described in [20], which also operates on the ternary valued image $g^{\mathrm{T},k}$. Basically, $c^{\mathrm{M},k}$ is obtained by applying a Short-Time Fourier transform to $g^{\mathrm{T},k}$ and then measuring the uniformity of the resulting Fourier coefficient magnitudes: highly skewed magnitude distributions correspond to structured regions, returning values of $c^{\mathrm{M},k}$ that are close to 1, while more uniform magnitude distributions correspond to unstructured regions, returning values close to 0.

The final matching score for candidate displacement $\boldsymbol{v}$ and location $\boldsymbol{n}$ is taken at the finest resolution level as $\rho^{\oplus,\rightarrow 0}(\boldsymbol{v})[\boldsymbol{n}]$.

IX. Search Procedure and Computational Complexity

In this section, we further explain the search procedure that is capable of generating a dense motion vector field with the same resolution as the frame used for motion estimation. This procedure is actually equivalent to the usual block-based motion estimation, except that it generates a dense motion field. We also discuss an efficient approximation of the direct implementation, exploiting the fact that the proposed 2-bit matching score tends to be a smooth, slowly varying function of spatial position and displacement. Furthermore, the computational complexity of the direct search procedure and the efficient approximation are shown and compared to that of block-based motion estimation with MSE as the criterion.

Conceptually at least, to find the dense motion field, we first displace frame f to obtain $f_{\boldsymbol{v}}$, for each candidate motion vector $\boldsymbol{v}$ within the search range. In the case where we use MSE in the original image domain as the matching score, $\rho(\boldsymbol{v})[\boldsymbol{n}]$ can be calculated as a moving average of the square of the difference between $g(\boldsymbol{n})$ and $f_{\boldsymbol{v}}(\boldsymbol{n})$ with window size $(2R+1)\times(2R+1)$. We compare $\rho(\boldsymbol{v})[\boldsymbol{n}]$ for each candidate motion vector $\boldsymbol{v}$, at each pixel location $\boldsymbol{n}$, to find a $\boldsymbol{v}$ that minimizes the matching score. This gives us the estimated motion vector field $\tilde{\mathcal{V}}[\boldsymbol{n}]$. The complexity of this dense motion estimation procedure is similar to that of traditional block-based motion estimation, with disjoint blocks and only one motion vector per block. In both cases, one multiplication and one subtraction are performed per pixel per displacement candidate; the coarse disjoint block approach requires an extra addition per pixel per displacement during the block accumulation phase, whereas the dense estimation procedure requires four additions per pixel per displacement for the moving average operation. It is also worth noting that the dense estimation procedure requires one comparison per pixel per displacement candidate, to discover $\tilde{\mathcal{V}}[\boldsymbol{n}]$. In summary, since comparison operations are comparable to adds, we can say that the dense MSE-based matching procedure has an overall complexity of (1 ×, 6 add) per pixel per displacement candidate.

For the proposed matching score $\rho^{\oplus,\rightarrow 0}(\boldsymbol{v})[\boldsymbol{n}]$, we again employ the moving average approach to find the estimated motion field $\tilde{\mathcal{V}}[\boldsymbol{n}]$. For each candidate $\boldsymbol{v}$, $f_{\boldsymbol{v}}^{k}$ has to be calculated from the Laplacian transform of $f_{\boldsymbol{v}}$, and $f_{\boldsymbol{v}}^{\oplus,k}$ calculated from $f_{\boldsymbol{v}}^{k}$. Moreover, the matching score calculated at each resolution level $\rho^{\oplus,\rightarrow k}(\boldsymbol{v})[\boldsymbol{n}]$ has to be interpolated and combined with the scores calculated at the next higher level. These operations add a certain amount of computational complexity compared to matching MSE in the image domain; however, the 2-bit matching procedure is simpler than MSE matching within any given resolution level. As it turns out, the complexity of this direct implementation is dominated by the calculation of $f_{\boldsymbol{v}}^{\oplus,k}$ for each candidate $\boldsymbol{v}$ and the interpolation of low resolution scores during the inter-resolution combination process. The former can be greatly reduced by observing that $f_{\boldsymbol{v}_1}^{\oplus,k}$ and $f_{\boldsymbol{v}_2}^{\oplus,k}$ are related by an integer shift whenever $\boldsymbol{v}_1-\boldsymbol{v}_2$ is a multiple of 2^{k-1}, so that the largest source of complexity eventually becomes that of interpolation.

For further efficiency improvements, we take advantage of the fact that $\rho^{\oplus,\rightarrow k}(\boldsymbol{v})[\boldsymbol{n}]$ tends to be a smooth, slowly varying function of both spatial position $\boldsymbol{n}$ and displacement $\boldsymbol{v}$, as suggested by Figure 8c. Specifically, we evaluate $\rho^{\oplus,\rightarrow k}(\boldsymbol{v})[\boldsymbol{n}]$ directly only for integer-valued locations $\boldsymbol{n}$ and displacements $\boldsymbol{v}$ whose coordinates are multiples of 2^{k-1}, after which the remaining values are approximated using a quadlinear interpolation procedure. This means that we only need to calculate $f^{\oplus,k}$ directly, since shifts of f by multiples of 2^{k-1} correspond to whole pixel shifts of $f^{\oplus,k}$. Our experiments show that, for a variety of test data, this quadlinear interpolation approximation has a negligible impact on the final estimated motion vector field $\tilde{\mathcal{V}}[\boldsymbol{n}]$.

When quadlinear interpolation is used, the cost of forming $f^{\oplus,k}$ can be largely ignored because it is independent of the number of displacements considered. At each level k, $\rho^{\oplus,k}(\boldsymbol{v}_{\boldsymbol{i}})[\boldsymbol{n}]$ must be evaluated for each location $\boldsymbol{n}$ and $\boldsymbol{v}=2^{k-1}\boldsymbol{i}$ such that $\boldsymbol{i}$ is an integer vector. This requires two XOR operations and an addition, plus four additions for computing the moving average. To generate $\rho^{\oplus,\rightarrow k}(\boldsymbol{v}_{\boldsymbol{i}})[\boldsymbol{n}]$ we also need to interpolate the matching scores from level $k+1$ and blend them with $\rho^{\oplus,k}(\boldsymbol{v}_{\boldsymbol{i}})[\boldsymbol{n}]$. Blending requires two additions and a multiplication, while the cost of quadlinear interpolation is α multiplications and 2α additions per output value, where $\alpha=\frac{15}{16}$. To see this, note that 1D linear interpolation by a factor of 2 requires two adds and one multiply to evaluate every second output sample; quadlinear interpolation may be performed one dimension at a time with each dimension produces half as many samples as the next one so that the total number of applications of the non-trivial interpolation operator is $\alpha=\frac{1}{2}+\frac{1}{4}+\frac{1}{8}+\frac{1}{16}$. Putting everything together, the cost of forming $\rho^{\oplus,\rightarrow k}$ is (2 XOR, $1+\alpha$ ×, $7+2\alpha$ add) for each sample location and motion candidate in level k. The total number of values which must be computed at level k is 16 times the number that must be computed at level $k+1$, from which we conclude that the overall complexity of the proposed method is $\frac{16}{15}=\frac{1}{\alpha}$ times that for level 0 alone. Noting that we also need a comparison operation (one add), the overall complexity can be expressed as ($2/\alpha$ XOR, $1/\alpha+1$ ×, $7/\alpha+3$ add) per output sample per displacement candidate.

The proposed method naturally yields half-pixel precision motion field, at a spatial resolution which is twice that of the original image. Based on our earlier analysis, the generation of a similar motion field using MSE as the matching score requires (1 ×, 6 add) per output sample per displacement candidate, which is less complex than the proposed method by a factor of perhaps 3, noting that multiplications are by far the most costly of the operations involved. Both methods can be adapted to generating motion fields with different densities and motion vector precisions, with similar implications for complexity.

X. Experimental Results

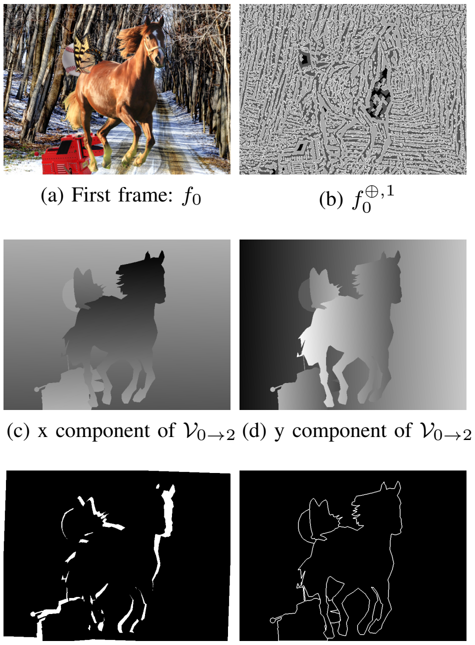
In this section, we first use synthetic test sequences, generated by composing moving sprites over a moving background image. The advantage of such content is that the true motion field $\mathcal{V}_{i\to j}[\boldsymbol{n}]$ between frames f_i and f_j is known. Moreover, the set of locations in f_i that are occluded in f_j, denoted as $\mathcal{O}_{i\to j}$ and the set of motion boundaries of f_i, denoted as $\mathcal{B}_i$, are all known. Motion boundary is defined as a 4-pixel wide region centred on the boundary of two different sprites. We made these test sequences available on the web at http://dstn.ee.unsw.edu.au/~rui. With this ground truth information, we can quantify the accuracy of the estimated motion field $\tilde{\mathcal{V}}_{i\to j}[\boldsymbol{n}]$ by calculating a quality measure E, defined as

$$E = \frac{1}{N}\sum_{\boldsymbol{n}}\frac{1}{2}\left\|\mathcal{V}[\boldsymbol{n}] - \tilde{\mathcal{V}}[\boldsymbol{n}]\right\|_1, \tag{8}$$

where N is the total number of pixels in a frame. This is simply the mean absolute difference of all x and y components in the motion field. We can also define $E_{\bar{\mathcal{O}}}$, as the average motion error excluding the occluded region,

$$E_{\bar{\mathcal{O}}} = \frac{1}{N - \|\mathcal{O}\|}\sum_{\boldsymbol{n}\notin\mathcal{O}}\frac{1}{2}\left\|\mathcal{V}[\boldsymbol{n}] - \tilde{\mathcal{V}}[\boldsymbol{n}]\right\|_1, \tag{9}$$

Similarly, we can define $E_{\bar{\mathcal{O}}\&\bar{\mathcal{B}}}$, where $\mathcal{B}$ is the motion boundary region defined earlier.



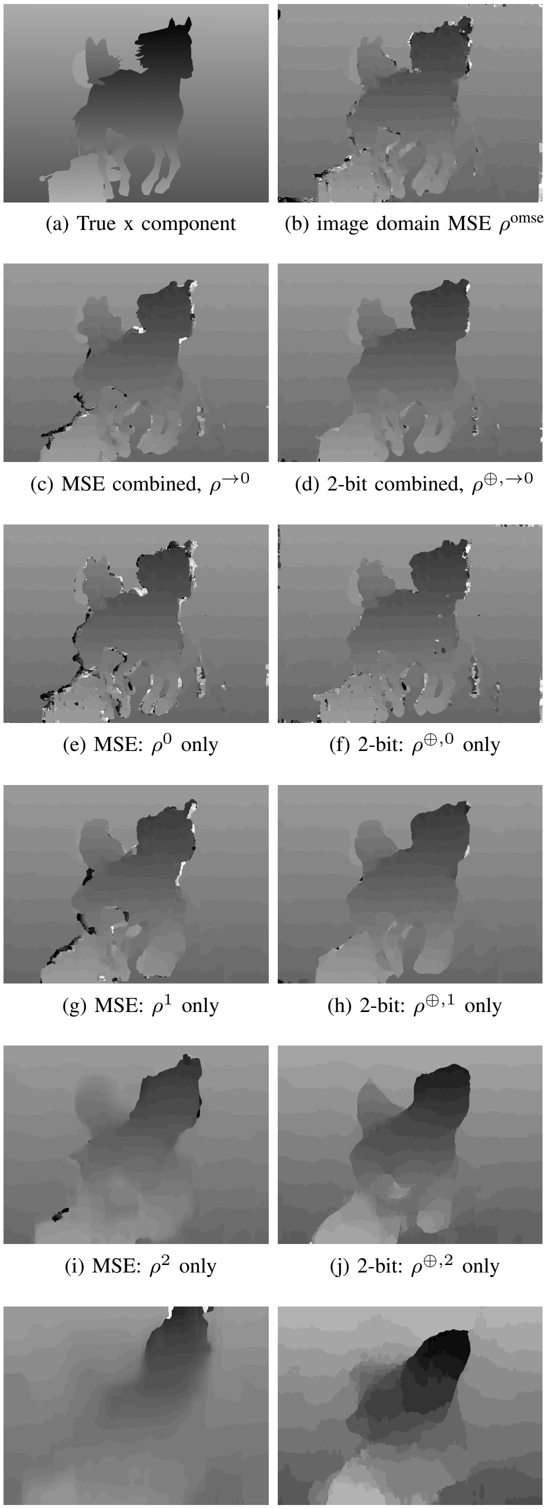
(a) First frame: f_0 (b) $f_0^{\oplus,1}$

(c) x component of $\mathcal{V}_{0\to 2}$ (d) y component of $\mathcal{V}_{0\to 2}$

(e) $\mathcal{O}_{0\to 2}$ (f) $\mathcal{B}_0$

Fig. 9: The first frame in the horse sequence of 10 frames and some ground truth associated with this frame. The frames are 640×480 in dimension. The motion in the sequence is constant and the maximum motion between two consecutive frames is 15 pixels. Only the luminance information is used for motion search. Motion estimates of the first two frames are shown in Figure 10.

We begin by considering the relatively complex test sequence shown in Figure 9, where four different sprites with translation and rotation are composed on a background image



(a) True x component (b) image domain MSE ρ^{omse}

(c) MSE combined, $\rho^{\to 0}$ (d) 2-bit combined, $\rho^{\oplus,\to 0}$

(e) MSE: ρ^0 only (f) 2-bit: $\rho^{\oplus,0}$ only

(g) MSE: ρ^1 only (h) 2-bit: $\rho^{\oplus,1}$ only

(i) MSE: ρ^2 only (j) 2-bit: $\rho^{\oplus,2}$ only

(k) MSE: ρ^3 only (l) 2-bit: $\rho^{\oplus,3}$ only

Fig. 10: Horizontal components of true motion $\mathcal{V}_{0\to 1}$ (in (a)) and motion estimates $\tilde{\mathcal{V}}_{0\to 1}$ (in the rest) for the first two frames of the horse sequence in Figure 9. It is shown as gray scale images (mid-gray represents 0, each gray image is displayed with the maximum contrast). 5 resolution levels are used in total, with a search range of ±20 at full pixel precision. The quality metrics E of the estimates are shown in Table I

TABLE I: Average motion errors for the horse sequence. The ranking of each method, measured relative to each of the performance measures E, $E_{\bar{\mathcal{O}}}$ and $E_{\bar{\mathcal{O}}\&\bar{\mathcal{B}}}$, are shown as subscripts in parentheses.

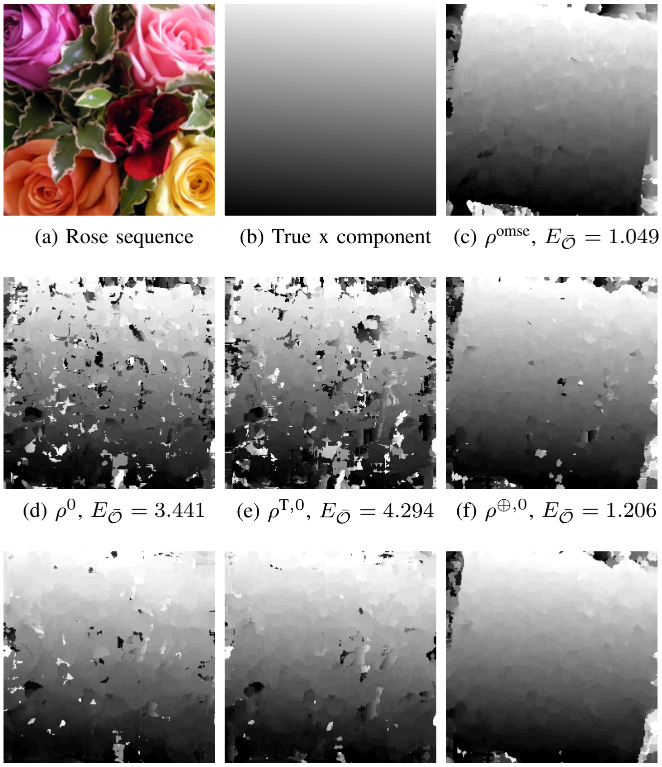
Matching Score	E	$E_{\bar{\mathcal{O}}}$	$E_{\bar{\mathcal{O}}\&\bar{\mathcal{B}}}$
ρ^{omse}	$0.781_{(7)}$	$0.620_{(6)}$	$0.527_{(6)}$
$\rho^{\rightarrow 0}$	$0.650_{(4)}$	$0.575_{(5)}$	$0.476_{(4)}$
$\rho^{\oplus,\rightarrow 0}$	$0.457_{(1)}$	$0.371_{(1)}$	$0.323_{(1)}$
ρ^{0}	0.851	0.735	0.588
$\rho^{\oplus,0}$	$0.653_{(5)}$	$0.494_{(3)}$	$0.438_{(3)}$
ρ^{1}	$0.741_{(6)}$	0.674	0.584
$\rho^{\oplus,1}$	$0.498_{(2)}$	$0.424_{(2)}$	$0.368_{(2)}$
ρ^{2}	0.853	0.793	0.733
$\rho^{\oplus,2}$	$0.607_{(3)}$	$0.563_{(4)}$	$0.502_{(5)}$
ρ^{3}	1.218	1.174	1.128
$\rho^{\oplus,3}$	0.907	0.889	0.829

with translation and rotation. We compare three different matching scores for motion estimation: the proposed 2-bit method; MSE in the original image domain; and a third measure formed by evaluating the MSE score separately on each resolution level and combining the results using Equation (7). These are denoted $\rho^{\oplus,\rightarrow 0}$, ρ^{omse} and $\rho^{\rightarrow 0}$. We also compare the performance of the proposed matching score and MSE within individual resolution levels k, writing $\rho^{\oplus,k}$ and ρ^{k} for these results. Note that the matching window size for ρ^{omse} is 15×15, the matching window size for $\rho^{\oplus,k}$ is explained in Section VII, and the matching window size for ρ^{k} is the same as $\rho^{\oplus,k}$.

Our results show that $\rho^{\oplus,k}$ produces more robust motion fields than ρ^{k} for all levels k. $\rho^{\oplus,\rightarrow 0}$ produces more robust motion fields than $\rho^{\rightarrow 0}$ and ρ^{omse}. The horizontal component of these various motion estimates are shown in Figure 10 for one particular pair of frames from the sequence. The average motion estimation errors over the entire sequence are reported in Table I in terms of E, $E_{\bar{\mathcal{O}}}$ and $E_{\bar{\mathcal{O}}\&\bar{\mathcal{B}}}$. These results are consistent with our early observations in [18], where test frames with different sprites and background images are used. Our results also show that an extra small non-zero noise threshold δ in the thresholding Equations (2), (3) and (4) has minimal effect on the overall performance of the algorithm. When $\delta = 1$ and $\delta = 4$ the motion error E is 0.458 and 0.490 respectively, comparing to 0.457 in Table I where no noise threshold δ is used. This is largely due to the ability of the weighting $c^{\mathrm{M},k}$ to distinguish noise and structure in the non-linear transform when combining matching scores from all resolutions.

In the next two subsections, we separately evaluate the performance of the proposed method under more restrictive conditions, so as to understand the features of the algorithm in greater detail. In the final subsection, we test the proposed approach on a standard optical flow dataset.

A. Performance with Global Rotation



(a) Rose sequence (b) True x component (c) ρ^{omse}, $E_{\bar{\mathcal{O}}} = 1.049$

(d) ρ^{0}, $E_{\bar{\mathcal{O}}} = 3.441$ (e) $\rho^{\mathrm{T},0}$, $E_{\bar{\mathcal{O}}} = 4.294$ (f) $\rho^{\oplus,0}$, $E_{\bar{\mathcal{O}}} = 1.206$

(g) $\rho^{\rightarrow 0}$, $E_{\bar{\mathcal{O}}} = 1.553$ (h) $\rho^{\mathrm{T},\rightarrow 0}$, $E_{\bar{\mathcal{O}}} = 1.323$ (i) $\rho^{\oplus,\rightarrow 0}$, $E_{\bar{\mathcal{O}}} = 0.889$

Fig. 11: (a) is the first frame in the rose sequence of 10 frames. The frames are 256×256 in dimension. A global motion of 10° rotation is introduced between consecutive frames. Only the luminance information is used for matching. (b) is the horizontal components of the true motion $\mathcal{V}_{0\rightarrow 1}$. The rest is the horizontal components of motion estimates $\tilde{\mathcal{V}}_{0\rightarrow 1}$ for the first two frames of the rose sequence shown in (a). It is shown as gray scale images (mid-gray represents 0, each gray image is displayed with the maximum contrast). 5 resolution levels are used in total, with a search range of ± 30 at full pixel precision. Note that $\bar{\mathcal{O}}$ excludes the regions that are occluded by the frame boundary. The quoted motion errors $E_{\bar{\mathcal{O}}}$ are averages within 10 consecutive frames.

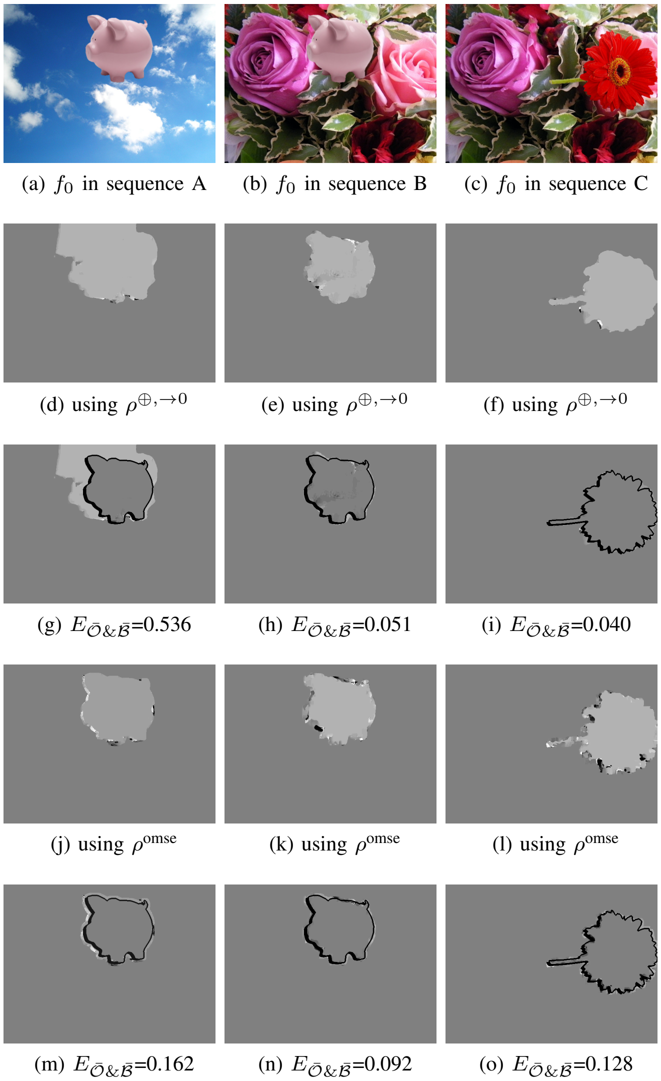
We now consider a simple test sequence shown in Figure 11a, where the only motion is a global rotation such that the frames contain no motion discontinuities. This corresponds to the analysis performed earlier in Section VI, where the two matching windows on two frames are related not only by translation but also rotation.

Our results show that the proposed method with dilation provides far better utilization of the high frequency subband information than without dilation (i.e. using $\rho^{\mathrm{T},\rightarrow 0}$ or setting $\epsilon = 0$) or using MSE as the matching score. The 2-bit final combined result $\rho^{\oplus,\rightarrow 0}$ is also superior to the image domain MSE approach ρ^{omse}. The actual horizontal estimated and ground truth motion fields are also shown in Figure 11.

One observation from this test sequence is that when f^{0} is rich in edge features, the motion estimates using $\rho^{\oplus,\rightarrow 0}$ are very similar to $\rho^{\oplus,0}$. This suggests that the combined result mainly comes from the highest resolution level. This also means low frequency effects such as illumination variation, would not affect the accuracy of the proposed approach for such content, unlike MSE-based approaches. However, such effects are not present in these experiments.

B. Performance with Motion Discontinuities

In this section, we deliberately introduce motion discontinuities by composing a single sprite on top of a stationary



(a) f_0 in sequence A (b) f_0 in sequence B (c) f_0 in sequence C

(d) using $\rho^{\oplus,\to 0}$ (e) using $\rho^{\oplus,\to 0}$ (f) using $\rho^{\oplus,\to 0}$

(g) $E_{\bar{\mathcal{O}}\&\bar{\mathcal{B}}}$=0.536 (h) $E_{\bar{\mathcal{O}}\&\bar{\mathcal{B}}}$=0.051 (i) $E_{\bar{\mathcal{O}}\&\bar{\mathcal{B}}}$=0.040

(j) using ρ^{omse} (k) using ρ^{omse} (l) using ρ^{omse}

(m) $E_{\bar{\mathcal{O}}\&\bar{\mathcal{B}}}$=0.162 (n) $E_{\bar{\mathcal{O}}\&\bar{\mathcal{B}}}$=0.092 (o) $E_{\bar{\mathcal{O}}\&\bar{\mathcal{B}}}$=0.128

Fig. 12: Each column shows horizontal components of motion estimates and differences between ground truth of two frames from one of the test sequences A, B and C. Each of the three sequences contains one sprite moving on a stationary background image and the sprite is moved 8 pixels to the left and 4 pixels down between consecutive frames. Sequence A represents a sprite with sparse texture on a background with sparse texture, Sequence B represents a sprite with sparse texture on a background with dense texture and Sequence C represents a sprite with dense texture on a background with dense texture. The improvement of the proposed method using $\rho^{\oplus,\to 0}$ over ρ^{omse} can be well observed in the motion estimates of sequence C. The first row contains frames from sequences of 10 frames which are 640×480 in dimension. The second row contains the results using the proposed 2-bit method $\rho^{\oplus,\to 0}$; the differences are shown in the third row. The fourth row contains results using ρ^{omse}; the differences are shown in the fifth row. All components and differences are shown as grey images (mid-gray represents 0, each gray image is displayed with the maximum contrast). In the difference image, the occluded and motion boundary region is labeled as black. The quoted motion errors $E_{\bar{\mathcal{O}}\&\bar{\mathcal{B}}}$ are averages within each sequence.

background. All motions are translational so that we can focus on the ability of the proposed algorithm to accurately identify the locations of motion discontinuities.

Our results show that when the regions of two different motions both contain sufficient edge features, the proposed 2-bit method $\rho^{\oplus,\to 0}$ is better at determining motion boundaries than image domain MSE ρ^{omse}. It has also been observed in [19] that motion estimation on overlapping blocks is able to resolve motion boundaries more accurately than what the block size may suggest. When one of the motion regions does not have sufficient edge features, the proposed method tends to propagate the motion from the other region to this region (from sprite to background or from background to sprite), while ρ^{omse} can utilize the low frequency information to determine motion at this smooth region. The horizontal components of the estimated motion fields on three test sequences with different amounts of edge features are shown in Figure 12. Figure 12 also shows the horizontal component differences between the estimates and the ground truth with $\mathcal{O}$ and $\mathcal{B}$ regions labeled.

C. Performance on Optical Flow Dataset

TABLE II: Motion error E of motion estimates on the Middlebury dataset using different matching scores. The ρ^{osad} column is matching SAD in the image domain; the "2-bit best" column is the best result selected from 2-bit method in Table III and 2-bit-abs method in Table IV; the "1-bit best" column is the best result selected from 1-bit method in Table V. The best performing method is labeled with a subscript "1" in parentheses.

Method	ρ^{osad}	ρ^{omse}	$\rho^{\oplus,\to 0}$	2-Bit Best	1-Bit Best
Dimetrodon	0.347	0.371	0.267	$0.265_{(1)}$	0.292
Grove2	0.568	0.640	0.343	$0.331_{(1)}$	0.352
Grove3	1.162	1.340	0.731	$0.651_{(1)}$	0.671
Hydrangea	0.350	0.384	0.198	$0.197_{(1)}$	0.201
RubberWhale	0.292	0.317	0.203	$0.202_{(1)}$	0.207
Urban2	1.229	1.425	0.937	$0.851_{(1)}$	0.961
Urban3	1.945	2.078	1.919	1.736	$1.652_{(1)}$
Venus	0.607	0.660	0.320	$0.297_{(1)}$	0.415

TABLE III: Motion error E of motion estimates on the Middlebury dataset using 2-bit method $\rho^{\oplus,\to 0}$ with different sizes of dilation kernel B_ϵ. The best performing dilation in each row is labeled with a subscript "1" in parentheses.

Method	$\epsilon = 0$	$\epsilon = 1$	$\epsilon = 2$	$\epsilon = 3$	$\epsilon = 4$
Dimetrodon	0.578	$0.265_{(1)}$	0.267	0.267	0.268
Grove2	0.335	$0.331_{(1)}$	0.336	0.343	0.353
Grove3	$0.651_{(1)}$	0.706	0.727	0.731	0.732
Hydrangea	0.200	$0.197_{(1)}$	$0.197_{(1)}$	0.198	0.198
RubberWhale	0.214	$0.202_{(1)}$	0.203	0.203	0.204
Urban2	1.042	$0.902_{(1)}$	0.919	0.937	0.950
Urban3	$1.777_{(1)}$	1.817	1.861	1.919	1.993
Venus	0.335	$0.297_{(1)}$	0.310	0.320	0.330

In this section, we test the proposed 2-bit method $\rho^{\oplus,\to 0}$, 2-bit-abs method and 1-bit method with different amount of dilation on a standard optical flow data set from Middlebury [17] and compare them with image domain MSE ρ^{omse} and image domain sum of absolute difference (SAD) ρ^{osad}.

Our results show that the proposed methods outperform image domain MSE and SAD on all test frames in the Middlebury dataset. This is shown in Table II. Our results also show that the dilation step almost always helps to improve the

TABLE IV: Motion error E of motion estimates on the Middlebury dataset using 2-bit-abs method with different sizes of dilation kernel B_ϵ. The best performing dilation in each row is labeled with a subscript “1” in parentheses.

Method	$\epsilon = 0$	$\epsilon = 1$	$\epsilon = 2$	$\epsilon = 3$	$\epsilon = 4$
Dimetrodon	0.637	$0.271_{(1)}$	0.274	0.277	0.278
Grove2	0.353	$0.337_{(1)}$	0.346	0.357	0.368
Grove3	$0.685_{(1)}$	0.693	0.741	0.763	0.769
Hydrangea	0.208	$0.202_{(1)}$	0.204	0.205	0.205
RubberWhale	0.220	$0.205_{(1)}$	0.206	0.207	0.207
Urban2	1.213	$0.851_{(1)}$	0.910	0.954	0.973
Urban3	1.844	$1.736_{(1)}$	1.792	1.873	1.953
Venus	0.383	$0.314_{(1)}$	0.337	0.348	0.353

TABLE V: Motion error E of motion estimates on the Middlebury dataset using 1-bit method with different sizes of dilation kernel B_ϵ. The best performing dilation in each row is labeled with a subscript “1” in parentheses.

Method	$\epsilon = 0$	$\epsilon = 1$	$\epsilon = 2$	$\epsilon = 3$
Dimetrodon	0.515	$0.292_{(1)}$	0.304	0.425
Grove2	$0.352_{(1)}$	0.452	0.939	3.401
Grove3	$0.671_{(1)}$	1.031	1.890	4.945
Hydrangea	$0.201_{(1)}$	0.242	0.604	2.369
RubberWhale	0.213	$0.207_{(1)}$	0.283	0.959
Urban2	1.311	$0.961_{(1)}$	1.654	4.315
Urban3	1.854	$1.652_{(1)}$	2.214	3.370
Venus	$0.415_{(1)}$	0.562	0.887	1.289

motion estimates for the 2-bit method, 2-bit-abs method and 1-bit method. For this particular dataset, having ϵ larger than 1 does not help to improve the result Further. We think this is perhaps because the dataset does not contain the amount of rotation that the chosen dilation kernel size $\epsilon = 3$ is trying to tolerate. This is shown in Table III, Table IV and Table V.

XI. Conclusion

We propose, analyse and evaluate a matching score on a class of non-linear transforms of the detail bands of a Laplacian pyramid, in a dense block-based full-search motion estimation setting. The non-linear transform involves a thresholding and a morphological dilation step. Our analysis shows that the proposed approach utilises high frequency subbands information better than MSE when the motion is non-translational, and provides guidance in choosing the parameters for the non-linear transforms and subsequent matching. In our experiments, the proposed approach is shown to produce more accurate motion estimation than MSE both in individual detail bands and in the combined multi-resolution matching score. This is especially true when the motion is non-translational, validating the analysis. The matching score from each detail band is combined using a local structural content estimator, which effectively gives a content-adaptive block size, addressing the well known aperture problem to some extent.

In addition, the proposed motion estimation method is capable of resolving motion boundaries more effectively than MSE. We hypothesise that this is because the thresholding process discards local contrast so that high contrast regions are less likely to dominate low contrast regions within the matching window.

We also describe an efficient implementation of the approach and show that its computational complexity is similar to the usual MSE block-based motion estimation.

Overall, this paper provides useful analysis and evaluation on a class of multi-resolution non-linear matching scores that may have broad applicability.

References

[1] A. Abdelazima, M. Varleya, and D. Ait-Boudaoudb, “Effect of the hadamard transform on motion estimation of different layers in video coding,” *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 38, 2010.

[2] A. Erturk and S. Erturk, “Two-bit transform for binary block motion estimation,” *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 15, no. 7, pp. 938–946, 2005.

[3] O. Urhan and S. Erturk, “Constrained one-bit transform for low complexity block motion estimation,” *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 17, no. 4, pp. 478–482, 2007.

[4] B. Natarajan, V. Bhaskaran, and K. Konstantinides, “Low-complexity block-based motion estimation via one-bit transforms,” *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 7, no. 4, pp. 702–706, 1997.

[5] P.-W. Wong and O. Au, “Modified one-bit transform for motion estimation,” *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 9, no. 7, pp. 1020–1024, 1999.

[6] K. Nam-Joon, S. Erturk, and L. Hyuk-Jae, “Two-bit transform based block motion estimation using second derivatives,” *Consumer Electronics, IEEE Transactions on*, vol. 55, no. 2, pp. 902–910, 2009.

[7] D. S.Taubman, “Method for providing motion-compensated multi-field enhancement of still images from video,” Apr.30, 2002 2002.

[8] S.-J. Ko, S.-H. Lee, S.-W. Jeon, and E.-S. Kang, “Fast digital image stabilizer based on gray-coded bit-plane matching,” *Consumer Electronics, IEEE Transactions on*, vol. 45, no. 3, pp. 598–603, 1999.

[9] A. Naman, R. Xu, and D. Taubman, “Inter-frame prediction using motion hints,” *20th Proc. IEEE Int. Conf. Image Proc. 2013*, September 2013.

[10] D. Tzovaras, M. G. Strintzis, and H. Sahinoglou, “Evaluation of multiresolution block matching techniques for motion and disparity estimation,” *Signal Processing: Image Communication*, vol. 6, no. 1, pp. 59–67, 1994.

[11] J. Chalidabhongse and C. C. J. Kuo, “Fast motion vector estimation using multiresolution-spatio-temporal correlations,” *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 7, no. 3, pp. 477–488, 1997.

[12] J. Magarey and N. Kingsbury, “Motion estimation using a complex-valued wavelet transform,” *Signal Processing, IEEE Transactions on*, vol. 46, no. 4, pp. 1069–1084, 1998.

[13] R. Mathew and D. Taubman, “Quad-tree motion modeling with leaf merging,” *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 20, no. 10, pp. 1331–1345, 2010.

[14] Y.-K. Tu, J.-F. Yang, Y.-N. Shen, and M.-T. Sun, “Fast variable-size block motion estimation using merging procedure with an adaptive threshold,” in *Multimedia and Expo, 2003. ICME ’03. Proceedings. 2003 International Conference on*, vol. 2, 2003, pp. II–789–92 vol.2.

[15] B. K. Horn and B. G. Schunck, “Determining optical flow,” *Artificial intelligence*, vol. 17, no. 1, pp. 185–203, 1981.

[16] B. D. Lucas and T. Kanade, “An iterative image registration technique with an application to stereo vision,” in *IJCAI*, vol. 81, 1981, pp. 674–679.

[17] S. Baker, D. Scharstein, J. Lewis, S. Roth, M. J. Black, and R. Szeliski, “A database and evaluation methodology for optical flow,” *International journal of computer vision*, vol. 92, no. 1, pp. 1–31, 2011.

[18] R. Xu and D. Taubman, “Robust dense block-based motion estimation using a two-bit transform on a laplacian pyramid,” *20th Proc. IEEE Int. Conf. Image Proc. 2013*, September 2013.

[19] M. T. Orchard and G. J. Sullivan, “Overlapped block motion compensation: An estimation-theoretic approach,” *Image Processing, IEEE Transactions on*, vol. 3, no. 5, pp. 693–699, 1994.

[20] A. Naman and D. Taubman, “A soft measure for identifying structure from randomness in images,” *20th Proc. IEEE Int. Conf. Image Proc. 2013*, September 2013.