Motion-Adaptive Transforms Based on the Laplacian of Vertex-Weighted Graphs

Du Liu and Markus Flierl

KTH Royal Institute of Technology

Stockholm, SE-10044, Sweden

{dul, mflierl}@kth.se

Abstract

We construct motion-adaptive transforms for image sequences by using the eigenvectors of Laplacian matrices defined on vertex-weighted graphs, where the weights of the vertices are defined by scale factors. The vertex weights determine only the first basis vector of the linear transform uniquely. Therefore, we use these weights to define two Laplacians of vertex-weighted graphs. The eigenvectors of each Laplacian share the first basis vector as defined by the scale factors only. As the first basis vector is common for all considered Laplacians, we refer to it as subspace constraint. The first Laplacian uses the inverse scale factors, whereas the second utilizes the scale factors directly. The scale factors result from the assumption of ideal motion. Hence, the ideal unscaled pixels are equally connected and we are free to form arbitrary graphs, such as complete graphs, ring graphs, or motion-inherited graphs. Experimental results on energy compaction show that the Laplacian which is based on the inverse scale factors outperforms the one which is based on the direct scale factors. Moreover, Laplacians of motion-inherited graphs are superior than that of complete or ring graphs, when assessing the energy compaction of the resulting motion-adaptive transforms.

1   Introduction

The objective of motion-adaptive transforms is to produce jointly coded frames for efficient video coding. For motion-connected pixels, we are able to perform linear transforms efficiently, as they are highly correlated. The well-known motion-compensated lifting wavelets permit a reversible filter structure, but struggle with unconnected, connected, and multi-connected pixels when performing the update step. Hence, [1] and [2] propose modified update operators to address this shortcoming. Instead of using a lifting wavelet, the class of motion-compensated orthogonal transforms (MCOT) is designed for successive pictures in a video sequence and maintains strict orthogonality for any motion field [3,4]. Not requiring the concept of lifting, these transforms compact the signal energy efficiently by using the concept of scale factors.

Formally, the signals can be defined on graphs for processing. For example, [5] considers lifting transforms on graphs based on spatial and temporal connections due to block motion. Further, spectral graph theory using the Laplacian matrix of graphs can be incorporated [6]. Utilizing the eigenvectors of the Laplacian matrix is helpful in many fields. For example, [7] uses these eigenvectors to define a discrete transform and to design a lowpass filter for smoothing curves and surfaces. Similarly, [8] constructs an edge-adaptive graph-based transform from the eigenvectors. Wavelets can be constructed based on the Laplacian eigenvectors as well [9]. Aspects of both signal processing and spectral graph theory are discussed in [10]. From the perspective of approximation theory, [11] argues that the Laplacian eigenbasis can be regarded as a meaningful Fourier transform.

In our previous work, a class of motion-adaptive transforms is proposed [12]. The transforms are constructed from vertex-weighted graphs, where the graphs are defined by the motion information only. The vertex weights of the graph determine one basis vector of the transform. The remaining basis vectors of the transform are derived from the covariance matrix which is represented in a dimension-reduced subspace. The transform is optimal in terms of energy compaction given the subspace constraint. However, the basis vectors derived from the reduced covariance matrix are signal dependent. Aiming at a transform basis that is independent of the signal, we designed a transform that is constructed by simply rotating the basis of the discrete cosine transform (DCT) [13]. However, since the subspace constraint determines only one basis vector, there are many ways to construct the higher basis vectors of the transform.

For our work, the energy compaction of the motion-adaptive transform is relevant. In this paper, we consider two definitions for the Laplacian matrix of the vertex-weighted graph. The eigenbasis of a Laplacian matrix serves as our transform, while the first basis vector meets our subspace constraint. Since the subspace constraint does not imply how the vertices are connected to each other, we are free to form graphs that are reasonable for the connected pixels. We also discuss the relation between the unweighted Laplacian matrix and our vertex-weighted Laplacian matrix. The resulting transform basis does not need to be transmitted, since the vertex-weighted graph is reconstructed from motion information at the decoder side.

The paper is organized as follows: Section 2 summarizes the subspace constraint as obtained from the vertex-weighted graph. Section 3 defines the Laplacian matrix of the vertex-weighted graph. Section 4 discusses the relation between the unweighted and the vertex-weighted Laplacian matrices. Section 5 presents the experimental results.

2　Subspace Constraint

In this section, we first introduce the scale factors, which are used to accommodate the energy compaction in orthonormal transforms [3]. The weights for vertex-weighted graphs are given by the scale factors. From the scale factors, we derive a subspace constraint that determines the transform in a one-dimensional linear subspace [12].

An orthonormal transform compacts the energy of pixels into a lowband coefficient. The magnitude of the lowband pixel intensity changes. If there are two motion-connected pixels x'_1 and x'_2, optimal energy compaction will produce a lowband coefficient $x_1 = (x'^2_1 + x'^2_2)^{\frac{1}{2}}$ and a highband coefficient of zero. On the other hand, ideal motion implies constant intensity for motion-connected pixels, which states $x'_1 = x'_2$. Assuming ideal motion, the lowband coefficient can be simplified to $x_1 = \sqrt{2}x'_1$ and the scale factor for the lowband is set to be $\sqrt{2}$. If there is a third motion-connected pixel, for ideal motion we have $x'_3 = x'_1$. Then, in the second step of the transform, the energy of the third pixel will be compacted to the lowband coefficient as well. As a result, the lowband coefficient becomes $x_1 = \sqrt{3}x'_1$. The scale factor is updated to $\sqrt{3}$, which captures the total energy.

Now we consider n motion-connected lowband pixels $\mathbf{x} = [x_1, x_2, \ldots, x_n]^T$. In general, we can express each lowband pixel as a product of the scale factor and the original pixel intensity, i.e., $x_k = c_k x'_k$, $k = 1, 2, \ldots, n$, where c_k is the scale factor and x'_k is the original intensity. If a pixel has not been transformed before, the scale factor is simply one. Now let $\mathbf{c} = [c_1, c_2, \ldots, c_n]^T$ denote a vector of these n scale factors. Similar to [3] and assuming ideal motion, the original pixel values are all equal, $x'_1 = x'_2 = \cdots = x'_n$. Letting T be an n-dimensional orthonormal transform, we want all the energy to be compacted into the first coefficient, i.e.,

$$\mathbf{y} = T^T \mathbf{x} = T^T \begin{bmatrix} c_1 x'_1 \\ c_2 x'_2 \\ \vdots \\ c_n x'_n \end{bmatrix} = \begin{bmatrix} \sqrt{\sum_{k=1}^n c_k^2} \cdot x'_1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \tag{2.1}$$

where $\mathbf{y} = [y_1, y_2, \ldots, y_n]^T$ is the output. As we see, the compaction of the energy is represented by $\sqrt{\sum_{k=1}^n c_k^2}$. Rewriting (2.1), for the first vector of the transform, we have

$$\mathbf{t}_1^T \mathbf{c} = \sqrt{\mathbf{c}^T \mathbf{c}}, \tag{2.2}$$

where $\mathbf{t}_1$ is the first vector in T. This implies that all the energy is captured by $\mathbf{t}_1$. Thus, $\mathbf{t}_1$ needs to be collinear with $\mathbf{c}$ such that all energy of $\mathbf{c}$ is captured by the basis vector $\mathbf{t}_1$. Consequently, we have

$$\mathbf{t}_k^T \mathbf{c} = 0, \ k = 2, 3, \ldots, n, \tag{2.3}$$

which implies that there is no energy left to be captured by $\mathbf{t}_2, \ldots, \mathbf{t}_n$, since they are orthogonal to $\mathbf{t}_1$. Hence, our first vector of the linear transform T is

$$\mathbf{t}_1 = \frac{\mathbf{c}}{\|\mathbf{c}\|_2} = \frac{\mathbf{c}}{\sqrt{\mathbf{c}^T \mathbf{c}}}. \tag{2.4}$$

If $x_1, x_2, \ldots, x_n$ are affected by noise, i.e., the original pixel values x'_k are not equal, perfect energy compaction as in (2.1) will not be achieved. However, we preserve the subspace constraint $\mathbf{t}_1 = \frac{\mathbf{c}}{\|\mathbf{c}\|_2}$ as it reflects the underlying motion.

3 Laplacian Matrices of Vertex-Weighted Graphs

The subspace constraint defines the first column of the transform. The remaining $n-1$ basis vectors are undetermined. To construct the remaining basis vectors, we are interested in a class of matrices whose matrix of eigenvectors is a candidate for T. Since graphs can be formed with the help of motion vectors, we use Laplacian matrices to represent our graphs. The vertices of our graph represent the motion-connected pixels, where the vertex weights are related to our scale factors.

In the following, we consider two possible definitions of the Laplacian matrix for a vertex-weighted graph. One is based on the inverse scale factors. The other relates to the scale factors directly [14]. In both cases, the resulting matrices of eigenvectors satisfy our subspace constraint. However, the main difference between the two definitions stems from the diagonal entries of the Laplacian matrix.

3.1 Laplacian for Vertex-Weighted Graphs Using Inverse Scale Factors

Let $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$ denote a graph with a set of n vertices $\mathcal{V}$ and a set of edges $\mathcal{E}$, where each vertex v_i is associated with the weight w_i for $i = 1, 2, \ldots, n$. Note this weight can be different from the scale factors, as it is only used to help defining the Laplacian matrix. Let D be the degree matrix with diagonal entries $d_1, d_2, \ldots, d_n$, where d_i represents the number of connections for vertex v_i. The unweighted Laplacian is

$$L^\circ(i,j) = \begin{cases} -1, & \text{if } (i,j) \in \mathcal{E} \\ d_i, & \text{if } i = j \\ 0, & \text{otherwise.} \end{cases} \tag{3.1}$$

We see that L° has an all-one eigenvector $\mathbf{1}$ and the corresponding eigenvalue zero. Now, let W be a diagonal weighting matrix with the ith diagonal entry w_i. The vertex-weighted Laplacian may be determined by

$$L^\star = W L^\circ W^T. \tag{3.2}$$

As discussed above, one eigenvector of the vertex-weighted Laplacian needs to satisfy our subspace constraint. The corresponding eigenvalue should be zero. From (3.2), we write the definition of the Laplacian of the vertex-weighted graph as

$$L^\star(i,j) = \begin{cases} -w_i w_j, & \text{if } (i,j) \in \mathcal{E} \\ d_i w_i^2, & \text{if } i = j \\ 0, & \text{otherwise.} \end{cases} \tag{3.3}$$

The definition states that if there is an edge between two vertices v_i and v_j, $L^\star(i,j)$ is $-w_i w_j$. The ith diagonal entry of $L^\star$ is $d_i w_i^2$, i.e. the degree multiplied by the squared weight. $L^\star(i,j)$ is zero if there is no edge. The unweighted Laplacian can be viewed as a special case with all weights equal to one.

From the definition, it can be shown that $L^\star$ has an eigenvector $W^{T-1}\mathbf{1}$. The corresponding eigenvalue is $\lambda_1^\star = 0$. Since we want the eigenvector matrix of $L^\star$ to be

our transform matrix T, the first eigenvector $W^{T^{-1}}\mathbf{1}$ should be collinear with $\mathbf{c}$, for simplicity, $W^T\mathbf{c} = \mathbf{1}$. Thus, $w_i = c_i^{-1}$. W becomes

$$W = \mathrm{diag}[w_1, w_2, \ldots, w_n] = \mathrm{diag}\left[c_1^{-1}, c_2^{-1}, \ldots, c_n^{-1}\right]. \tag{3.4}$$

This guarantees that the eigenvector matrix of $L^\star$ satisfies the subspace constraint. Note also the special case where the number of connections and the square of the scale factor at all vertices v_i are equal, i.e., $d_i = c_i^2$, then the vertex-weighted Laplacian becomes a normalized Laplacian with all diagonal elements equal to one.

Given a vector $\mathbf{x} \in \mathbb{R}^n$ defined on the graph, the unweighted Laplacian quadratic form is $\mathbf{x}^T L^\circ \mathbf{x} = \sum_{(i,j)\in\mathcal{E}}(x_i - x_j)^2$. It measures the smoothness of $\mathbf{x}$. If $\mathbf{x}$ is smooth, this value will be small. If $\mathbf{x}$ varies significantly over any edge, it will be large. Similarly, the vertex-weighted Laplacian quadratic form is

$$\mathbf{x}^T L^\star \mathbf{x} = \mathbf{x}^T W L^\circ W^T \mathbf{x} = \sum_{(i,j)\in\mathcal{E}} (w_i x_i - w_j x_j)^2 = \sum_{(i,j)\in\mathcal{E}} \left(\frac{x_i}{c_i} - \frac{x_j}{c_j}\right)^2. \tag{3.5}$$

We see that $L^\star$ is positive semi-definite. (3.5) implies that using the weighted Laplacian is equivalent to scaling the signal down with scale factors before computing the unweighted Laplacian.

3.2 Laplacian for Vertex-Weighted Graphs Using Direct Scale Factors

The second definition for the Laplacian is proposed in [14]. We denote it as $\check{L}$. Similarly to the eigenvector $W^{T^{-1}}\mathbf{1}$ of $L^\star$, this Laplacian $\check{L}$ has an eigenvector that can be obtained directly from the vertex weights as well. Assume each vertex v_i is associated with a weight $\check{w}_i$, $i = 1, 2, \ldots, n$. Again, $\check{w}_i$ is only used to define the Laplacian matrix. $\check{L}$ is given by

$$\check{L}(i, j) = \begin{cases} -\check{w}_i \check{w}_j, & \text{if } (i, j) \in \mathcal{E} \\ \sum_{(i,k)\in\mathcal{E}} \check{w}_k^2, & \text{if } i = j \\ 0, & \text{otherwise.} \end{cases} \tag{3.6}$$

We see that the diagonal entries of $L^\star$ and $\check{L}$ are different. The ith diagonal entry for $L^\star$ is $d_i w_i^2$. While the ith diagonal entry for $\check{L}$ is the summation of all squared weights whose vertices are connected to the vertex v_i.

Let $\check{W}$ be a diagonal matrix with the ith diagonal entry $\check{w}_i$. It is easy to show that $\check{L}$ has one eigenvector $\check{W}\mathbf{1}$ with the corresponding eigenvalue $\check{\lambda}_1 = 0$. Again, we need this eigenvector to be collinear with $\mathbf{c}$. Thus, $\check{w}_i = c_i$. Comparing to W, instead of having the ith diagonal entry as c_i^{-1}, $\check{W}$ simply has c_i, i.e.,

$$\check{W} = \mathrm{diag}[\check{w}_1, \check{w}_2, \ldots, \check{w}_n] = \mathrm{diag}\left[c_1, c_2, \ldots, c_n\right]. \tag{3.7}$$

The Laplacian quadratic form is

$$\mathbf{x}^T \check{L}\mathbf{x} = \sum_{(i,j)\in\mathcal{E}} (\check{w}_i x_j - \check{w}_j x_i)^2 = \sum_{(i,j)\in\mathcal{E}} (c_i x_j - c_j x_i)^2. \tag{3.8}$$

This implies that for each pair $(i, j) \in \mathcal{E}$, the signal values x_i and x_j are scaled by the index-switched scale factors c_j and c_i, respectively.

4 Unweighted and Vertex-Weighted Laplacian Matrices

The ideal motion in Sec. 2 assumes that the n connected pixels are $x'_1 = x'_2 = \cdots = x'_n$. It implies that these n pixels are equally important. There is no further assumption about which pixel should be connected to which other pixels. Since the subspace constraint does not imply the structure of the graph, the connections between the pixels can be defined freely without violating the subspace constraint. As these n pixels are connected together in a group, one intuitive option is a complete graph, i.e., each pixel is connected to all other pixels. An example of a complete graph with $n = 4$ is depicted in Fig. 1(a), where the vertices are labeled x'_1 to x'_4. Another option can be a ring graph, i.e., each vertex is connected to two neighbors, as shown in Fig. 1(b). The complete graph or the ring graph does not emphasize any vertex, as each vertex is equally connected. The Laplacian matrix for an unweighted complete graph or an unweighted ring graph is circulant, thus the discrete Fourier transform (DFT) diagonalizes the Laplacian matrix. A third way to form a graph is to inherit the connections from motion vectors. This is appealing, since the motion vector trajectories can be viewed as graphs as well. As exemplified in Fig. 1(c), x'_1 is the motion reference pixel for x'_2, and x'_2 is the motion reference pixel for x'_3 and x'_4.
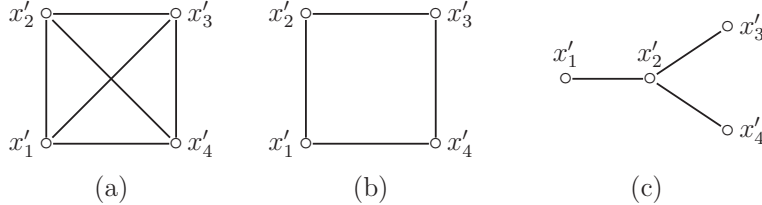

| | | |
| (a) | (b) | (c) |

Figure 1: Examples of graphs. (a) A complete graph. (b) A ring graph. (c) A motion-inherited graph.

In the following, we study the relationship between vertex-weighted and unweighted Laplacian matrices in the context of our subspace constraint. Let F be the eigenvector matrix of the unweighted Laplacian and Λ° the eigenvalue matrix, i.e., $L^\circ = F\Lambda^\circ F^H$, where $(\cdot)^H$ denotes Hermitian transpose. Similarly, let T be the eigenvector matrix of $L^\star$ with a corresponding eigenvalue matrix $\Lambda^\star$. From Sec. 3 we know that the eigenvector matrix T always satisfies the subspace constraint $\mathbf{t}_1$, either for a complete graph, a ring, or an inherited graph. The eigenvector $\mathbf{t}_1$ is fixed, while the remaining eigenvectors are not unique. They are determined by the structure of the graph and the definition of the Laplacian matrix.

Further, let the weighting between the eigenvalue matrices be $\Lambda^\star = U\Lambda^\circ U^T$, where U is a diagonal matrix with the ith nonzero diagonal entry u_i. Since both L° and $L^\star$ are singular, i.e., $\lambda_1^\circ = 0$ and $\lambda_1^\star = 0$, the value of u_1 is not unique. Based on the eigen-decompositions, we have the following relation

$$L^\star = T\Lambda^\star T^H = TU\Lambda^\circ U^T T^H = TUF^H L^\circ \left(TUF^H\right)^H = GL^\circ G^H, \qquad (4.1)$$

where $G = TUF^H$ is expressed by a singular value decomposition. On the other hand, the relation between L° and $L^\star$ can also be expressed by the weighting process $L^\star = WL^\circ W^T$, as shown in (3.2). This challenges the characterization of the relationship.

To obtain a characteristic relation between L° and $L^\star$, we modify the two matrices such that they become nonsingular while maintaining all eigenvectors. We assume small positive eigenvalues ε_1° and $\varepsilon_1^\star$ to replace λ_1° and $\lambda_1^\star$, respectively. Let $\mathbf{f}_k$ be the kth eigenvector of L° and $\mathbf{t}_k$ the kth eigenvector of $L^\star$. We use the following nonsingular matrices to replace the Laplacian matrices, i.e.,

$$\widetilde{L}^\circ = L^\circ + \varepsilon_1^\circ \mathbf{f}_1 \mathbf{f}_1^H = \varepsilon_1^\circ \mathbf{f}_1 \mathbf{f}_1^H + \sum_{k=2}^{n} \lambda_k^\circ \mathbf{f}_k \mathbf{f}_k^H \qquad (4.2)$$

and

$$\widetilde{L}^\star = L^\star + \varepsilon_1^\star \mathbf{t}_1 \mathbf{t}_1^H = \varepsilon_1^\star \mathbf{t}_1 \mathbf{t}_1^H + \sum_{k=2}^{n} \lambda_k^\star \mathbf{t}_k \mathbf{t}_k^H. \qquad (4.3)$$

Clearly, this modification does not change the eigenvectors in F and T. Let $\widetilde{\Lambda}^\circ$ and $\widetilde{\Lambda}^\star$ be the eigenvalue matrices of $\widetilde{L}^\circ$ and $\widetilde{L}^\star$, respectively. Then, $\widetilde{\Lambda}^\circ = \mathrm{diag}[\varepsilon_1^\circ, \lambda_2^\circ, \ldots, \lambda_n^\circ]$, $\widetilde{\Lambda}^\star = \mathrm{diag}[\varepsilon_1^\star, \lambda_2^\star, \ldots, \lambda_n^\star]$, and $u_1^2 = \varepsilon_1^\star / \varepsilon_1^\circ$. With that, we have a characteristic relation

$$\widetilde{L}^\star = T\widetilde{\Lambda}^\star T^H = TU\widetilde{\Lambda}^\circ U^T T^H = TUF^H \widetilde{L}^\circ \left(TUF^H\right)^H = G\widetilde{L}^\circ G^H, \qquad (4.4)$$

where G is again TUF^H. Since both T and F are unitary, the eigen-decomposition for GG^H and $G^H G$ are

$$GG^H = TU^2 T^H \quad \text{and} \quad G^H G = FU^2 F^H, \qquad (4.5)$$

respectively. As U has full rank, G is nonsingular. T and F can be expressed as $T = GFU^{-1}$ and $F = G^{-1}TU$, respectively. Then, the ith vector of T can be computed as

$$\mathbf{t}_i = \frac{G\mathbf{f}_i}{\|G\mathbf{f}_i\|_2}, \quad \text{for} \quad i = 1, 2, \ldots, n. \qquad (4.6)$$

Fig. 2 depicts a commutative diagram for $\widetilde{L}^\circ$, $\widetilde{L}^\star$, F, and T. The problem of mapping F to T is related to the mapping from $\widetilde{L}^\circ$ to $\widetilde{L}^\star$. The link is given by the eigenvalue decomposition / Karhunen Loeve Transform (KLT). This commutative diagram shows that there exists a unique T for each graph.

Figure 2: Commutative diagram for the modified Laplacian matrices and transforms.

5   Experimental Results

In the experiments, we evaluate the energy compaction for the QCIF sequences *Fore-man*, *Mobile* and *City*. We compare the Motion-Compensated Orthogonal Transform (MCOT) [3], the Graph-Based Rotation (GBR) [13], the eigenvectors of the Laplacian $L^\star$ with inverse scale factors and complete graphs (VW-ISF-CG), ring graphs (VW-ISF-RG), and inherited graphs (VW-ISF-IG), as well as the eigenvectors of the Laplacian $\check{L}$ with direct scale factors and inherited graphs (VW-DSF-IG) [14].

The MCOT is performed along the connections as given by motion vectors where each transform is operated on two coefficients. The GBR is obtained by rotating the discrete cosine transform (DCT) basis. We concluded in [12] that $\mathbf{t}_1$ approximates the energy compaction of the KLT well, since it compacts lowband energy close to that of the KLT. It confirms the relevance of the subspace constraint. It is shown in [13] that the GBR compacts highband energy close to optimal, given the subspace constraint. Here, the GBR is provided as a reference for comparison purpose.

We perform transforms on a hierarchical decomposition of a group of pictures (GOP) with a size of sixteen. A GOP includes two hierarchical decomposition levels. In the first level, sixteen frames are separated into four graphs sets. Each set contains four successive pictures. In the second level, four tempotal lowbands from the four sets are grouped together for process. The GOP size and graph sets can be changed, as the vertex-weighted Laplacian can represent a random graph structure. The graphs are defined by 16×16 block motion with a search range of ± 32.

Tables 1 to 3 show the relative energy in the temporal subband frames on the second decomposition level for *Foreman*, *Mobile*, and *City*, respectively. We utilize all the energy in the test frames. Since the transforms are orthonormal, the energy is preserved by the transform. All transforms contain the same basis vector $\mathbf{t}_1$, thus, the same lowband energy is produced.

For the remaining basis vectors $\mathbf{t}_k$, $k > 1$, we see that the second largest energy component of the VW-ISF-IG is greater than that of the VW-ISF-CG and the VW-ISF-RG, which implies that the inherited graph is better than the complete graph and the ring graph. It has higher energy than the second largest energy component of the GBR as well. In Table 3, we observe almost 1% improvement compared to the GBR. However, the second largest energy components of the VW-ISF-CG and VW-ISF-RG are both lower than that of the GBR, which means these two kinds of graphs are less efficient in energy compaction.

Then, we compare the inherited graph for two different definitions of the Laplacian matrix. We see that the second largest energy component of the VW-ISF-IG is larger than that of the VW-DSF-IG, which indicates that the Laplacian with inverse scale factors works better than the one with direct scale factors. We also observe that VW-DSF-IG still provides better energy compaction than the GBR. Thus, utilizing the inherited graph is efficient in terms of energy compaction.

The smallest energy component is also given by the VW-ISF-IG. For example, in Table 3, the smallest energy component of VW-ISF-IG is about 0.5% lower than that of the GBR and about 1.5% lower than that of the MCOT. Therefore, among the proposed transforms, VW-ISF-IG achieves the highest energy compaction.

Table 1: Relative energy in the temporal subband frames on the second decomposition level for *Foreman*.

	Low	High 1	High 2	High 3
MCOT	97.51%	0.74%	1.00%	0.75%
GBR	97.51%	1.03%	0.73%	0.73%
VW-ISF-CG	97.51%	0.88%	0.73%	0.88%
VW-ISF-RG	97.51%	0.83%	0.96%	0.70%
VW-ISF-IG	97.51%	**1.13%**	0.81%	**0.55%**
VW-DSF-IG	97.51%	1.08%	0.82%	0.59%

Table 2: Relative energy in the temporal subband frames on the second decomposition level for *Mobile*.

	Low	High 1	High 2	High 3
MCOT	92.39%	3.07%	2.73%	1.81%
GBR	92.39%	4.66%	1.67%	1.28%
VW-ISF-CG	92.39%	1.97%	2.06%	3.58%
VW-ISF-RG	92.39%	2.92%	2.97%	1.72%
VW-ISF-IG	92.39%	**4.72%**	1.85%	**1.04%**
VW-DSF-IG	92.39%	4.71%	1.84%	1.06%

Table 3: Relative energy in the temporal subband frames on the second decomposition level for *City*.

	Low	High 1	High 2	High 3
MCOT	82.18%	5.24%	6.96%	5.62%
GBR	82.18%	7.16%	4.61%	6.05%
VW-ISF-CG	82.18%	5.75%	6.00%	6.07%
VW-ISF-RG	82.18%	5.94%	6.05%	5.83%
VW-ISF-IG	82.18%	**8.14%**	5.50%	**4.18%**
VW-DSF-IG	82.18%	7.79%	5.39%	4.64%

6 Conclusions

In this paper, we consider Laplacian matrices of vertex-weighted graphs, where the weights of the vertices are defined by scale factors. The vertex-weighted graph determines uniquely the first basis vector of the linear transform. We refer to this as the subspace constraint. The vertex-weighted Laplacian matrices are defined such that the eigenvector matrices satisfy this subspace constraint. The eigenvector matrix defines our motion-adaptive transform for image sequences. We compare Laplacian matrices for different weights and different graphs. The results show that using a motion-inherited graph Laplacian which is defined by inverse scale factors is efficient in terms of energy compaction.

7 Acknowledgments

References

[1] C. Tillier, B. Pesquet-Popescu, and M. van der Schaar, "Improved update operators for lifting-based motion-compensated temporal filtering," *IEEE Signal Processing Letters*, vol. 12, no. 2, pp. 146–149, 2005.

[2] B. Girod and S. Han, "Optimum update for motion-compensated lifting," *IEEE Signal Processing Letters*, vol. 12, no. 2, pp. 150–153, 2005.

[3] M. Flierl and B. Girod, "A motion-compensated orthogonal transform with energy-concentration constraint," in *Proc. of the IEEE International Workshop on Multimedia Signal Processing*, Oct. 2006, pp. 391–394.

[4] ——, "Half-pel accurate motion-compensated orthogonal video transforms," in *Proc. of the IEEE Data Compression Conference*, Mar. 2007, pp. 13–22.

[5] E. Martínez-Enríquez, F. Díaz-de-María, and A. Ortega, "Video encoder based on lifting transforms on graphs," in *Proc. of the IEEE International Conference on Image Processing*, Sept. 2011, pp. 3509–3512.

[6] F. R. K. Chung, *Spectral Graph Theory*, ser. CMBS Regional Conference Series in Mathematics, No. 92.   American Mathematical Society, 1997.

[7] G. Taubin, "Curve and surface smoothing without shrinkage," in *Proc. of the International Conference on Computer Vision*, June 1995, pp. 852–857.

[8] W.-S. Kim, S. K. Narang, and A. Ortega, "Graph based transforms for depth video coding," in *Proc. of the IEEE International Conference on Acoustics, Speech and Signal Processing*, Mar. 2012, pp. 813–816.

[9] D. K. Hammond, P. Vandergheynst, and R. Gribonval, "Wavelets on graphs via spectral graph theory," *Applied and Computational Harmonic Analysis*, vol. 30, no. 2, pp. 129–150, 2011.

[10] D. I. Shuman, S. K. Narang, P. Frossard, A. Ortega, and P. Vandergheynst, "The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains," *IEEE Signal Processing Magazine*, vol. 30, no. 3, pp. 83–98, 2013.

[11] X. Zhu and M. Rabbat, "Approximating signals supported on graphs," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2012, pp. 3921–3924.

[12] D. Liu and M. Flierl, "Motion-adaptive transforms based on vertex-weighted graphs," in *Proc. of the IEEE Data Compression Conference*, Mar. 2013, pp. 181–190.

[13] ——, "Graph-based rotation of the DCT basis for motion-adaptive transforms," in *Proc. of the IEEE International Conference on Image Processing*, Sept. 2013, pp. 1802–1805.

[14] F. R. K. Chung and R. P. Langlands, "A combinatorial Laplacian with vertex weights," *Journal of Combinatorial Theory, Series A*, vol. 75, no. 2, pp. 316–327, 1996.