Video Completion Using Bandlet Transform

Ali Mosleh, *Student Member, IEEE*, Nizar Bouguila, *Senior Member, IEEE*, and A. Ben Hamza, *Senior Member, IEEE*

***Abstract*—In this paper, we address the video completion problem for two general cases: 1) filling-in the missing regions of videos captured by a non-stationary camera, and 2) filling-in the missing part of video sequences recorded by a stationary camera. For each case, a novel video completion technique based on the bandlet transform is presented. In the first case, a priority-based exemplar algorithm, which applies the bandlet transform and its generated coefficients along with motion information, is used to fill-in the occluded moving object or the removed region. In the second case, our proposed method is followed by a foreground/background segmentation preprocessing step to generate moving objects and background frames in order to facilitate the video completion task. The technique fills-in the background frames after removing objects by means of a precise optimization in the bandlet transform domain. Then, the occluded part of a moving object is completed by a priority-based algorithm which applies frames' geometry properties using the bandlet transform. Our experimental results indicate that the proposed video completion technique maintains both the spatial and temporal consistency and also demonstrate the effectiveness of the bandlet transform in video completion.**

***Index Terms*—Bandlets, inpainting, motion, video completion.**

I. Introduction

VIDEO completion is the task of automatically filling-in the missing parts in video sequences caused by removal of undesired objects or scratches on frames. Due to its powerful ability to restore damaged digital videos, video completion has attracted great attention in the past few years. Research works in this field can be broadly grouped into two main categories: inpainting and texture synthesis. Inpainting methods employ geometric algorithms, mostly based on partial differential equations (PDE). These methods address the inpainting problem as an interpolation task to estimate the pixel flows and propagate them from the boundary of a missing part into its center. The methods of the second category are exemplar-based techniques that search for the best texture match in the whole undamaged sequence, and then propagate it into the missing parts. This task is performed in such a way that it maintains both the temporal and spatial consistency.

Manuscript received September 02, 2011; revised December 19, 2011, March 09, 2012, and April 30, 2012; accepted May 01, 2012. Date of publication May 10, 2012; date of current version November 22, 2012. This work was supported in part by NSERC Discovery grants. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Feng Wu.

A. Mosleh is with the Department of Electrical and Computer Engineering, Faculty of Engineering and Computer Science, Concordia University, Montréal, QC H3G 2W1 Canada (e-mail: mos_ali@encs.concordia.ca).

N. Bouguila and A. B. Hamza are with Concordia Institute for Information Systems Engineering, Faculty of Engineering and Computer Science, Concordia University, Montreal, QC H3G 2W1 Canada (e-mail: bouguila@ciise.concordia.ca; hamza@ciise.concordia.ca).

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

Digital Object Identifier 10.1109/TMM.2012.2198802

Video completion is distinguished from still image inpainting by its challenging task of handling a large dimensionality and maintaining a reasonable spatio-temporal consistency. Image inpainting methods cannot be directly applied in video completion tasks due to the crucial importance of visual consistency. As an example, the 2-D tensor voting image inpainting approach in [1] generates shadow and ghost effects if applied directly for videos. As another example of an effective image inpainting technique, the method introduced in the two-part paper [2], [3] adaptively determines small magnitude coefficients of a signal transform and predicts the missing region considering that these coefficients are small. Although this recovery method is not limited to only inpainting spatial signals, it is very challenging to adapt it to video completion tasks dealing with unsound and damaged estimated motion vectors which are missing in the case of general video completion tasks. Thus, image inpainting techniques must be adapted so that they preserve motion continuation in video completion. A leading digital inpainting approach is [4] which employs nonlinear PDEs to perform image and video frame inpainting to imitate what artists manually do to fix old pictures. A concise review of mathematical models designated for PDE and interpolation based image inpainting methods is presented in [5]. As a modification of the method introduced in [4], a precise and efficient solution which applies fluid dynamic Navier-Stoke equations is developed in [6]. The digital inpainting task based on PDEs is followed in [7], which derives a third-order PDE based on Taylor expansion to propagate the border isophotes to the missing regions. Also, another video inpainting method is introduced in [8], which benefits from discrete *p-Laplacian* regularization on a weighted graph. Unlike the majority of video inpainting techniques, this method does not perform frame-by-frame inpainting but considers the whole video sequence as a volume to do the inpainting in all the frames. A general framework that models the regularity with a 3-D vector field to extract spatio-temporal regularities is introduced in [9]. Although this model is not specialized for video inpainting purposes, it shows promising results for this application. However, these video inpainting methods are suitable only for completion of thin damaged regions.

Due to the success of the exemplar-based image completion method proposed in [10], texture-based video completion approaches have tried to adapt this technique to the temporal properties of video frames. In this method, a correct order of filling-in process leads to a high performance in the completion task. In fact, the priority of a missing region to be filled is defined essentially by *confidence* and *data* values. A combination of these two values results in a proper criterion to find the high curvature and reliable structures. In the case of video sequences, important properties, such as availability, trackability and motion vectors of the pixels, edge properties, and geometric properties contribute to calculate the priority of the missing regions to be filled-in. For

instance, the method introduced in [11], which is extended in [12] for camera motion containing sequences, performs moving object segmentation to separate the background and foreground of the video to reduce the search space for completion of partially occluded moving objects. In this method a motion confidence value is used to find the priority of the filling-in area in order to maintain the temporal consistency in the foreground completion task. For the background completion step, the image inpainting technique introduced in [10] is adopted. A similar approach with modifications on data term to find the best priority is introduced in [13], based on analysis of continuities on stationary and non-stationary videos with slow or fast foreground motions. This technique is further improved in [14] for various camera motions. The work introduced in [15] also fills-in the missing regions by applying a priority-based scheme. The priority is determined based on the trackability of the pixels. After selecting the highest priority fragment around the boundary of the missing region, it is completed using a graph cut fragment updating instead of copying just a similar texture model from the undamaged region. Another method which considers the whole video as a volume (3-D space) has been introduced in [16]. This priority-based method considers the video completion task as a global optimization to search for the best match, with a well-defined objective function. Another algorithm, based on a motion layer segmentation, is proposed in [17]. Each separate layer is completed using the image inpainting method then all the layers are combined in order to restore the final video. However, this method does not consider the temporal consistency of adjacent frames. The technique in [18] performs video completion in the motion and spatial domains separately. The method benefits from a global motion estimation scheme well to track patches containing missing regions in the adjacent frames. This method is specialized for damaged digitized vintage films by an illumination regularization step. This method maintains spatio-temporal consistency very well due to the applied motion estimation and illumination regularization. Roughly speaking, these methods are quite similar to each other, and each one tries to calculate the priority of filling-in order, based on appropriate spatial and/or temporal properties of the available data.

Object-based methods can be considered as a sub-category of the texture synthesis based video completion methods. In [19] an object-based approach is proposed to deal with stationary camera videos. The interesting part of this method is the foreground completion that utilizes object templates. A fixed size sliding window to include a set of continuous object templates is defined. An insufficient number of postures in the database results in a low performance of the completion. Another object-based technique is introduced in [20]. The target video is decomposed to color and illumination frames using a manual layer segmentation method. Spatio-temporal and also illumination domain consistency is maintained by means of tensor voting. Then, the occluded object is reconstructed by synthesizing other available objects. In [21] a scheme is proposed to derive a virtual contour of the occluded object. This contour is suitable to find a good replacement for the occluded object in the postures set of the video.

We introduce a totally different video completion approach by applying the bandlet transform. In this paper, our earlier work [22] is improved for general cases. The method benefits from advantages of both inpainting and exemplar-based approaches to complete video sequences after object removal. The video completion problem is addressed for two general cases: 1) completion of videos recorded with a non-stationary camera and 2) completion of video sequences captured with a stationary camera. In both cases, the effective feature of the bandlets, i.e., revealing the geometric details of the surface of images (frames), is employed.

In the first case, a priority-based video completion method, which applies geometric properties of the patches using the bandlet transform, is carried out. This method fills-in the occluded parts of the frames after object removal by finding the best match in the available pixels in the whole video. The priority of the missing area to be filled-in first is determined by the amount of reliable data and also the largest motion vectors around the missing area. Hence, a motion estimation algorithm is required besides the bandlet transform to find the motion vectors of each frame block. Then, a patch matching scheme in the bandlet coefficient domain is effectively done to find the best candidate to replace the missing part.

In the case of dealing with video frames captured with a static camera, the problem is addressed by removal of the moving object or a portion of the background and filling-in the remaining hole, followed by completion of the partially occluded moving objects. The basic requirement of the proposed video completion method, in this case, is the separation of the moving objects from the background. Therefore, a bilayer video segmentation [23] is performed to separate the foreground and background of a video sequence. Firstly, the remaining hole after object removal is filled-in by temporally available data, then an optimization is performed in the bandlet transform domain in order to inpaint the video background. Secondly, for the completion of the partially occluded part of the moving object, another priority-based method is performed which also applies the patches' geometric properties revealed by means of the bandlet transform.

The rest of this paper is organized as follows: an overview of the bandlet transform is presented in Section II. Then, the proposed bandlet-based video completion method is discussed in Section III. In Section IV, the experimental results are shown. Finally, the paper is concluded in Section V.

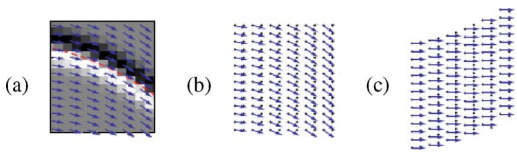
II. Bandlet Transform

In images, geometric regularity along edges is an anisotropic regularity. However, conventional wavelet bases only exploit the isotropic regularity on square domains of varying sizes. An image can be differentiable in a direction parallel to the tangent of an edge curve even though, very likely, the image may be discontinuous across the contour. Fig. 1(a) shows the geometric flow in the direction of edges in the hat of Lenna. Such an anisotropic regularity is exploited by the bandlet transform which constructs orthogonal vectors that are elongated in the direction of the maximum regularity of the function. Hence, the bandlet transform is considered as an effective tool to capture the geometric properties of an image.

The first bandlet bases were introduced in [24] and [25] having optimal approximations for geometrically regular

(a) (b)

Fig. 1. (a) Geometric flows in the direction of edges. (b) Image geometric segmentation.



(a) (b) (c)

Fig. 2. (a) Wavelet coefficients and geometric flow. (b) Geometric flow and sampling position. (c) Warped sampling [28].

functions. Then, the bandlet bases have been improved by a multi-scale geometry defined over the coefficients of a wavelet basis [26], [27]. In this section only a brief overview of bandlets is provided. For more details, the reader is referred to [28].

A. Orthogonal Bandlets Approximation

The polynomial approximation by means of a thresholding in an orthogonal Alpert basis is computed for the bandlet approximation. The Alpert transform can be considered as a polynomial wavelet transform adapted to an irregular sampling grid. It is obtained by orthogonalizing multi-resolution space of polynomials defined on the irregular sampling grid. An example of such sampling grid is shown in Fig. 2(c). The resulting vectors have vanishing moments on this irregular sampling grid, which is the basic need to approximate the warped wavelet coefficients. A few vectors from Alpert basis can efficiently approximate a vector corresponding to a sampling of a function with an anisotropic regularity. This kind of bandletization of wavelet coefficients is done by an Alpert transform defines a set of bandlet coefficients. These coefficients can be written as inner products $\langle f, b^k_{j,l,n}\rangle$ of the original image f with bandlet functions that are linear combinations of wavelet functions

$$b^k_{j,l,n}(x) = \sum_p a_{l,n}[p]\psi^k_{j,p}(x) \tag{1}$$

where the $a_{l,n}[p]$ are the coefficients of the Alpert transform. These coefficients depend on the local geometric flow, since this flow defines the warped sampling locations, as it is exemplified in Fig. 2(c). The bandlet function is defined at some location n and wavelet scale 2^j. Another scale factor $2^l > 2^j$ is introduced by the Alpert transform which defines the elongation of the bandlet function. Also, the bandlet inherits the regularity of the mother wavelets ψ^k_j.

B. Geometric Flow Segmentation Approximation

The family of orthogonal bandlets depends on the local adapted flow over small squares for each scale 2^j and orientation k. This parallel flow is characterized by an integral curve, such as the depicted one as the dashed red plot in Fig. 2(a). Due to the bidimensional regularization performed by the smoothing of the wavelet ψ^k_j, i.e., $f_i = f * \psi^k_j$, this integral curve does not need to be strictly parallel to the contour.

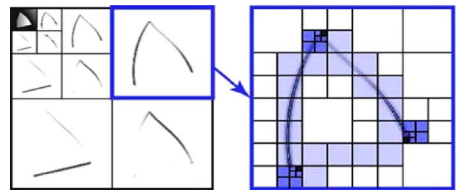


Fig. 3. Example of quadtree segmentation on scales of the wavelet transform of an image [28].

One needs to segment the set of wavelet coefficients in squares S, in order to approximate the geometry by a polynomial flow. This segmentation is obtained for each scale 2^j and orientation k of the wavelet transform using a recursive subdivision in dyadic squares of various sizes. This subdivision results in a quadtree that specifies if a square S should be further subdivided in four sub-squares with twice smaller size or not. There is no geometric directional regularity to exploit, if there is no specific direction of regularity inside a square. This is the case either in uniformly regular regions or at the vicinity of edge junctions. Thus, it is not necessary to modify the wavelet basis. A sample of such segmentation is shown in Fig. 3. Obviously, only for the edge squares, the adaptive flow is required to be computed in order to produce the bandlet bases which exploit the anisotropic regularity of an image. Through scales the geometric structures of an image evolves. Therefore, a different geometry Γ^k_j can be chosen for each scale 2^j and orientation k. The set of all geometries is noted as $\Gamma = \{\Gamma^k_j\}$ that consists of all the adapted flows of the quadtree segmentation squares.

In our work, due to the high complexity of quadtree segmentation, we employ fixed size squares instead of dynamically finding the size of each square. Fig. 1(b) shows the result of finding the geometric flow for each same-sized square of the quadtree, on the finest scale of the wavelet transform.

III. Bandlet-Based Video Completion

The video completion task is considered for two general video sequences: 1) video sequences containing camera motions and 2) video sequences captured by a stationary camera. In fact, two different techniques are introduced here for these two cases. In both cases, the bandlet transform has a vital role in filling-in the missing area in the video frames.

A. Completion of Motion Camera Produced Videos

The video sequence may have some missing data due to noise or any kind of damage, or maybe an object or a part of the background in the video sequence is occluded by the target object desired to be removed. Therefore, after removing the unwanted object, the missing parts of the video frames need to be completed. This task is carried out by means of a priority-based exemplar method, which applies the bandlet transform effectively. The priority of a filling-in process has an important role in enhancing the performance of video completion. The goal is to give a high priority to the missing area Ω on the border $\partial\Omega$,

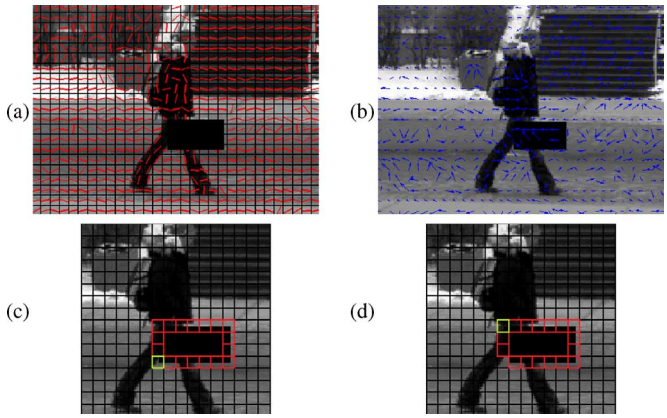


Fig. 4. Priority calculation. (a) Bandlet-based defined geometry. (b) BMA motion vectors. (c) Highest priority border block. (d) Repaired border block and the next highest priority border block.

surrounded by more available data, and also by strictly moving pixels. After finding which area should be filled-in first, we search for the best match in the whole sequence to complete the missing region.

1) Computing Filling-In Priority: An exemplar-based video completion algorithm works well if it gives higher priority to the regions that lie around enough reliable data and also moving pixels. Satisfaction of this fact leads to visually pleasant completion results. Therefore, the filling-in priority of the holes in the entire frames is computed using *structure confidence* and *motion confidence* values. The structure confidence value indicates the reliable pixels around each patch of the missing area border, and the motion confidence value indicates the strictly moving parts in the vicinity of the missing region. In order to compute these two values, the conventional 2-D patch generating is not required. Since the bandlet coefficients are needed in later stages, we can use the already generated bandlet squares to generate patches in our priority computation step. Fig. 4(a) shows the segmentation squares of the bandlet transform of a damaged frame.

To compute these two values for each part of the missing area border $\partial\Omega$, the bandlet squares (blocks) $p_{\partial\Omega}$ of the border are determined. Then, the structure confidence value for each $p_{\partial\Omega}$ is computed as

$$C(p_{\partial\Omega}) = \frac{\sum_{q\in\Psi_{p_{\partial\Omega}}} C(q)}{|\Psi_{p_{\partial\Omega}}|} \tag{2}$$

where $\Psi_{p_{\partial\Omega}}$ is a 3-D patch consisting neighboring blocks centered at $p_{\partial\Omega}$ whose size is $|\Psi_{p_{\partial\Omega}}| = N_x \times N_y \times N_t$ in term of number of blocks (N_x, N_y, and N_t indicate number of blocks along directions x, y, and t, respectively). This value is conceptually similar to the confidence value introduced in [10] adapted to videos in [12]. However, one needs to notice that we calculate it for generated bandlet blocks, not the pixels. During the initialization process, the structure confidence value is set to 1 for the blocks p in the source (available) region Φ and 0 for the blocks in the missing area Ω, i.e., $C(p) = 0\ \forall p \in \Omega$ and $C(p) = 1\ \forall p \in \Phi$. This structure confidence value can be considered as a measure of reliable data around each block of the missing region border. In other words, a block $p_{\partial\Omega}$ on $\partial\Omega$, with undamaged blocks around it, has a larger amount of confidence than those of other border patches.

In order to preserve the temporal consistency, the completion process should continue with the regions containing remarkable motions. Thus, higher priorities should be assigned to the areas in the vicinity of pixels with strict motions. We applied a simple 1-pixel precision version of a well-known block matching motion estimation (BMA), which is applied in MPEG-1 standard [29] to capture the motion vectors of the frame blocks of the whole sequence. Fig. 4(b) depicts the estimated block motion vectors for the bandlet squares of a sample frame.

For each $p_{\partial\Omega}$, the motion confidence value is computed as

$$M(p_{\partial\Omega}) = \frac{\sum_{q\in\Psi_{p_{\partial\Omega}}} D(q)}{|\Psi_{p_{\partial\Omega}}| \times \max(D_{BMA})} \tag{3}$$

where $|\Psi_{p_{\partial\Omega}}| = N_x \times N_y \times N_t$ is the total number of neighboring blocks centered at block $p_{\partial\Omega}$. $D(q)$ is the motion magnitude of the block q, i.e., $\sqrt{D_x^2(q) + D_y^2(q)}$. The parameters $D_x(q)$ and $D_y(q)$ represent the motion vector of block q considering x and y directions. These two parameters are found using the BMA. $\max(D_{BMA})$ represents the maximum possible estimated motion magnitude. For instance, in a BMA with a search range of 8 blocks, $\max(D_{BMA})$ is $\sqrt{8^2 + 8^2}$.

Finally, these two parameters are combined to obtain the priority value of each border pixel:

$$P(p_{\partial\Omega}) = C(p_{\partial\Omega}) \times M(p_{\partial\Omega}). \tag{4}$$

This value gives a high priority to the missing region having substantial available data nearby and also highly moving pixels. Fig. 4(c) shows the highest priority border block, illustrated in green, to be filled-in first for a single frame. This choice is reasonable since the indicated block lies at the corner of the missing area that has the most possible available data around and also close to the moving person's leg which has a strict motion relatively to the other parts of this frame. Note that this block is the center of a 3-D patch $\Psi_{p_{\partial\Omega}}$ contains spatial and temporal neighboring blocks.

2) Searching for the Best Match: Once the high priority block $p_{\partial\Omega}$ on the missing area border is found, a search on all the source region frames is carried out to find the best match for the 3-D patch $\Psi_{p_{\partial\Omega}}$. The center block of the found patch is substituted with $p_{\partial\Omega}$ to fill-in the missing area of the border. This is performed using the bandlet coefficients matching by means of sum of squared differences (SSD) through all the patches in all the source regions:

$$\acute{\Psi}_p = \arg\min_{\Psi_q\in\Phi_f} [SSD_B(\Psi_{p_{\partial\Omega}}, \Psi_q) \times SSD_C(\Psi_{p_{\partial\Omega}}, \Psi_q)] \tag{5}$$

where $\acute{\Psi}_p$ is the best match found for the patch $\Psi_{p_{\partial\Omega}}$. In (5), Φ_f indicates the whole available source regions, in all the video frames and SSD_B is defined in the bandlet transform domain as

$$SSD_B(\Psi_{q_1}, \Psi_{q_2}) = \sum_{(j,l,k,n,x)} \|B1^k_{j,l,n}(x) - B2^k_{j,l,n}(x)\|^2 \tag{6}$$

where $B^k_{j,l,n}(x)$ indicates the produced bandlet coefficients using the bandlet basis in (1) for the corresponding pixels x in Ψ_{q_1} and Ψ_{q_2}. SSD_B is an efficient measure to find the patches with very similar structure in the frames since geometry

properties of each frame in the video sequence is summarized very well with local clustering of similar geometric vectors in the form of bandlet coefficients.

Since the bandlet coefficients are generated from the gray-scale version of each frame, the color information needs to be considered in the patch matching process. Therefore, we define SSD_C as follows and employ it in (5):

$$SSD_C(\Psi_{q_1}, \Psi_{q_2}) = \sum_{(x,y,t)} \|\Psi_{q_1}(x,y,t) - \Psi_{q_2}(x,y,t)\|^2 \quad (7)$$

where, for each pixel at position (x, y) in the frame t within the patch, we have a 3-D vector (R, G, B).

Having found the proper source patch $\acute{\Psi}_p$, the value of the border block, $p_{\partial\Omega}$, is replaced by the corresponding block, p, which is the center block in the obtained source patch ($p \in \acute{\Psi}_p$). Then, the structure confidence value should be updated to give a lower priority to the blocks that just filled a part of the hole; therefore the confidence value is updated as

$$\acute{C}(p_{\partial\Omega}) = \alpha C(p_{\partial\Omega}) \quad (8)$$

where α is a weight factor $0 < \alpha \leq 1$ for the previous structure confidence value $C(p_{\partial\Omega})$. This simple update reduces the confidence value for the newly found block, which is, indeed, not reliable. Therefore, the priority of the region near the newly filled-in area is reduced. In this way, the filling-in process is performed around the border instead of in the missing area's center. Also, the motion vectors $D_x(p)$, $D_y(p)$ of the found block p are assigned to the border block $p_{\partial\Omega}$. Moreover, bandlet transform is applied to obtain new bandlet coefficients of the frame for the next step's block matching process. All the steps of this completion algorithm to fill-in the missing area are shown in Algorithm 1. The result of this algorithm for the highest priority block on the missing area of the frame shown in Fig. 4(c) is illustrated in Fig. 4(d). It is worth mentioning that the search and patch matching at each step is carried out over all the frames. It means that at each step, the highest border patches can be in different frames and the patch matching is not performed frame-by-frame in order to avoid flickering effect in the results.

Algorithm 1: Bandlet-based motion video completion.

1: Initialize confidence value for all the blocks; $C(p) = 0 \ \ \forall p \in \Omega$ and $C(p) = 1 \ \ \forall p \in \Phi$

2: Apply the bandlet transform on the whole sequence

3: Find the border blocks, $p_{\partial\Omega}$, for the frames

4: Compute priority of all the $p_{\partial\Omega}$, using (4)

5: Find the best match, $\acute{\Psi}_p$, for the highest priority border patch, $\Psi_{p_{\partial\Omega}}$, centered at $p_{\partial\Omega}$ in the frames using (5)

6: Replace the block $p_{\partial\Omega}$ with the corresponding block p

7: Update the structure confidence value using (8)

8: $D_x(p_{\partial\Omega}) = D_x(p)$, $D_y(p_{\partial\Omega}) = D_y(p)$

9: Generate new bandlet coefficients for the repaired frame

10: If there is any $p_{\partial\Omega}$, then go to step 4; else stop

B. Completion of Static Camera Videos

In the case of dealing with video sequences captured with static cameras, video completion cannot be carried-out well using the algorithm introduced in Section III-A since this algorithm is dependant on the global motion in the priority calculation. In addition, static camera videos can benefit from background/foreground separation. The background/foreground segmentation is important since we can detect and remove the moving object in the case of filling-in background. Moreover, we have the available data of the moving object to fill-in the missing area of the occluded object. This task is done by means of a bilayer video segmentation method proposed in [23] that employs motion, color, as well as contrast together, unlike other methods that only apply motion or color/contrast alone, to perform segmentation.

Foreground and background frames are generated after the bilayer segmentation process. Then, two cases are considered: firstly, the holes created in the background frames after object removal need to be filled-in; secondly, the removed object might have occluded another moving object. Hence, it is necessary to fill-in the occluded parts of the moving object. As follows, these two cases are discussed independently and for each case a solution is introduced using the bandlet transform. As mentioned before, the motivation behind applying bandlets stems from the fact that the geometry of each frame in the video sequence is summarized very well with local clustering of similar geometric vectors.

1) Background Inpainting: One important task of video completion is to fill-in the holes produced after object removal. In this case, we first find the available corresponding pixels of the holes to fill-in the missing part. Then, the final remaining hole in all the frames is completed by an inpainting method using bandlets.

The process is done on the background frames of the sequence. The missing portion in each frame is denoted by Ω. Then, a global search is performed in the background frames to find the corresponding pixel of the missing pixel in Ω of the current frame. To maintain the consistency, the temporally nearest pixel is selected to be placed in the current frame's Ω. If no temporal information is available for the current missing pixel, the process will continue with another pixel in Ω. This stage is performed on all the missing parts of the frames until a fixed hole remains in the entire sequence. This hole, existed in all the frames, will be inpainted in the next stage.

The inpainting problem is formulated as follows. An image I contains some missing pixels that belong to Ω and the goal is to find an image $\acute{I}$ such that $\acute{I}(x)$ is equal to $I(x)$ for the pixels that belong to the source ($\Phi = I \setminus \Omega$) area, i.e., $\acute{I}(x) = I(x) \ \forall x \notin \Omega$ and also the overall geometry of the new image $\acute{I}$ is supposed to have the same geometrical regularity as the original image I in Φ. These conditions can be enforced in the bandlet image representation [28]. By minimizing the ℓ^1 norm of the bandlet image representation, we can achieve a solution for the inpainting problem:

$$\acute{I} = \arg\min_{g,\Gamma} \sum_{v} |B^{\Gamma}_{g_v}|, \quad \forall x \notin \Omega, \ \ \acute{I}(x) = I(x) \quad (9)$$

where Γ is the bandlet derived geometry of an image and B_{g_v} denotes the bandlet coefficients of the corresponding geometry

Γ of the image g. B_{g_v} is found by inner product of bandlet bases of (1) and g. This minimization is done by adopting the *soft thresholding* algorithm, which has been used already for other multi-scale image representations such as wavelets [30], instead of an exhaustive optimization process.

Assuming the overall geometry to be fixed, the thresholding scheme, done iteratively, minimizes (9) by updating the estimate $I^{(i)}$ as follows:

$$I^{(i+1)} = T_\lambda^B(\acute{I}^{(i)}) \tag{10}$$

$$\acute{I}^{(i)} = \begin{cases} y(x) & \text{if } x \notin \Omega \\ I^{(i)}(x) & \text{if } x \in \Omega \end{cases} \tag{11}$$

where $y(x)$ denotes the pixels of the original image. In (10), T_λ^B is the soft threshold which is performed on the bandlet domain of the image $\acute{I}^{(i)}$ defined as

$$T_\lambda^B(g) = \sum_{j,l,k} t_\lambda(\langle g, b_{j,l,k}\rangle).b_{j,l,k} \tag{12}$$

where $t_\lambda(x) = \max((|x| - \lambda)/|x|, 0)x$, and $b_{j,l,k}$ indicate the bandlet bases in the scales and orientations of j, l, and k (Section II). The value of λ is decreased toward 0 during the iterations.

Algorithm 2: Bandlet-based background inpainting algorithm.

1: $i = 0$ and $I^{(i=0)} = y$

2: Find $\acute{I}^{(i)}$ using (11)

3: Apply bandlet transform on $\acute{I}^{(i)}$ (Section II)

4: Update the estimate; $I^{(i+1)} = T_\lambda^B(\acute{I}^{(i)})$ (applying (12) on the bandlet coefficients of $\acute{I}^{(i)}$ and performing inverse bandlet transform to generate $I^{(i+1)}$)

5: $i \leftarrow i + 1$, if $|I^{(i+1)} - I^{(i)}| < \varepsilon$ stop, else go to step 2

The entire minimization algorithm using soft thresholding is detailed in Algorithm 2, where the iterations end if the difference of two consecutive obtained estimates become lower than a small value ε. Fig. 5 shows some of the results of the inpainting process on an image at various iterations. The original image in Fig. 5(a) is scratched in Fig. 5(b), then the inpainting is applied to remove the scratches (Ω). The starting value of λ along with the assigned size of bandlet squares directly affects the degree of smoothness in the inpainting results. A larger value for λ results in a fewer number of iterations for Algorithm 2 since it reduces the difference of two consecutive image estimates. At the same time, a large λ results in a higher degree of smoothness in the final result. On the other hand, a small starting value of λ generates better results but increases number of iteration steps. Fig. 6 shows the inpainting results of a damaged image using various values for λ.

Once having found a fixed hole in all the video frames after object removal and temporally filling-in the frames, one can use this inpainting approach to complete the hole in one frame and simply propagate the results to all the other frames.

2) Moving Foreground Completion: A moving object may be occluded by the target object desired to be removed. Therefore, after removing the unwanted object, the occluded parts of the moving object need to be completed. The whole process, in this case, is only performed on the foreground frames because the foreground and background have been already separated. This filling-in process is also based on priority of the missing area to be filled and a patch matching procedure. In this case, the goal is to give high priority to the missing area on the border $\partial\Omega$, surrounded by more available data, and also high curvature textures. In contrast to the algorithm introduced in Section III-A, the priority computation does not need to consider motion confidence, since we already have only the moving pixels of the frames, not any static part.

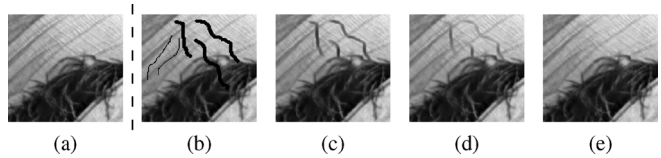


Fig. 5. Inpainting iterations. (a) Original image. (b) Scratched images. (c–e) Sample inpainting iterations.

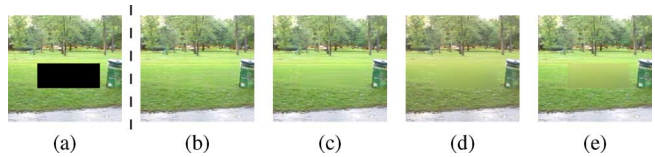


Fig. 6. Inpainting results with various λ values. (a) Damaged image. (b) $\lambda = 1$. (c) $\lambda = 3$. (d) $\lambda = 5$. (e) $\lambda = 10$.

Computing Filling-In Priority: The foreground exemplar-based completion algorithm can work well if it gives higher priority to the regions that lie on the continuation of image structures. Hence, filling-in priority of the hole in each frame is computed using the *confidence* and *data* values. The confidence value indicates the available pixels around each pixel of the missing area border [10] which is conceptually, similar to (2). The data value indicates the geometric structure in the vicinity of the missing region. To compute these two values for each pixel of the border $\partial\Omega$, a patch Ψ_p centered at a pixel p on $\partial\Omega$ is determined. Then, the confidence value for a pixel p is computed as follows:

$$C(p) = \frac{\sum_{q \in \Psi_p \cap \Phi} C(q)}{|\Psi_p|} \tag{13}$$

where $|\Psi_p|$ is the size of the 3-D patch Ψ_p and Φ is the source area ($I \setminus \Omega$). The confidence value is set to 1 for the pixels in the source (available) region and 0 for the pixels in the missing area, i.e., $C(p) = 0\ \forall p \in \Omega$ and $C(p) = 1\ \forall p \in \Phi$ in the initialization step. Note that here the confidence is determined for each pixel, not the blocks as done in (2). As pointed out in Section III-A, the confidence value can be considered as a measure of reliable data around each pixel of the missing area border. In Fig. 7(a), the pixels with the highest and the lowest confidence on the missing area border are depicted as the center of, respectively, a green and a red square patch.

The computation of the data value depends on the structure of the texture around each patch. Since the bandlet transform summarizes the geometry property of the texture, we use it in this part of computation. The bandlet transform is performed on

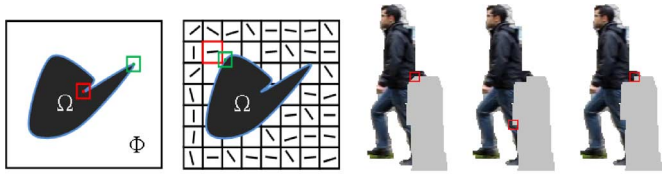


Fig. 7. Occluded moving object completion of motion camera videos. (a) Patches with the highest and lowest confidence. (b) Data value computation. (c) From left to right, three consecutive iterations of completion algorithm.

all the foreground frames to obtain the geometry of the neighboring region of each patch. Then, the nearest dyadic segmentation square of the bandlet transform, which lies in the Φ region of the frame to the desired patch, is found using the Euclidian distance. Fig. 7(b) depicts the nearest bandlet square to a patch centered at the border of the missing area. The geometry property of the found square, indicated as the angle of the dominant edge in the bandlet square, is employed to obtain the data value for each pixel of the missing area border:

$$D(p) = \begin{cases} \frac{G_s}{\beta} & \text{if } 0 \le G_s < 90 \\ 2 - \frac{G_s}{\beta} & \text{if } 90 \le G_s < 180 \end{cases} \tag{14}$$

where G_s is the bandlet based geometry property of the dyadic square in the source region, which is the nearest square to the patch Ψ_p. In fact, G is the bandlet-based geometry Γ but in a form that can be measured by degree. Since, the geometry values range in $[0, 180)$, a proper value for the normalization value β, can be 90.

The data value is a reliable measure to give priority to the highly structured textures, i.e., high curvature regions to be filled-in first. Then the priority value of each border pixel is obtained by

$$P(p) = C(p) \times D(p). \tag{15}$$

This measure gives priority to the missing areas having high curvature texture and having substantial available data nearby.

Algorithm 3: Bandlet-based moving object completion.

1: Initialize confidence value;

$$C(p) = 0 \ \forall p \in \Omega \text{ and } C(p) = 1 \ \forall p \in \Phi$$

2: Apply the bandlet transform on the foreground frames

3: Find the border pixels, $\partial\Omega$, for all the frames

4: Compute priority of $\partial\Omega$ pixels using (15) $\forall p \in \partial\Omega$

5: Find the best match, $\acute{\Psi}_p$, for the highest priority border patch, Ψ_p, in all the foreground frames using (16)

6: Replace $p \in \Omega \cap \Psi_p$ with pixel $\acute{p}$ such that $\acute{p} \in \acute{\Psi}_p$

7: Update the confidence value $\acute{C}(p) = \alpha C(p) \ \ \forall p \in \Omega \cap \Psi_p$

8: Generate a new bandlet geometry for the frames containing the 3-D patch Ψ_p

9: If there is any $\partial\Omega$ pixel, then go to step 4; else stop

Searching for the Best Match: Once the high priority patch Ψ_p on the missing area border is found, a search on all the source region foreground frames is carried out to find the best patch to be used to fill-in the missing area of the border patch. This is performed using the SSD through all the patches in all the source regions:

$$\acute{\Psi}_p = \arg \min_{\Psi_q \in \Phi_f} SSD(\Psi_q, \Psi_p) \tag{16}$$

where $\acute{\Psi}_p$ is the best match found for the patch Ψ_p. In (16), Φ_f indicates the whole available source regions, in all the foreground frames, and the SSD is defined as the color vector SSD_c in (7).

Having found the proper source patch $\acute{\Psi}_p$, the value of each missing pixel, $p \in \Psi_p \cap \Omega$, is replaced by the corresponding pixel in the obtained source patch $\acute{p} \in \acute{\Psi}_p$. Then, the confidence value should be updated as in (8), but here for the pixels, not the blocks, to give a lower priority to the pixels which already filled a part of the hole. In this way, the filling-in process is performed around the border instead of in the missing area's center. Also, the bandlet transform is applied to obtain the new geometric properties of the frame to compute data values for the next step. All the steps of the completion algorithm to fill-in the partially occluded moving objects are shown in Algorithm 3. Fig. 7(c) shows the first three iterations of this completion method on one single frame. The highest priority border patch is shown with a red square at each iteration. Once the moving object is completed, the new generated foreground frames are combined with the background frames to produce the final completed video sequence.

IV. Experimental Results

Our video completion method is tested on several video sequences regarding the aforementioned general cases: non-stationary camera and stationary camera produced videos. The videos can be found on http://users.encs.concordia.ca/~mos_ali/video_completion/tmm.htm. Also, completion results of the proposed method are compared with two state-of-the-art approaches presented in [12] and [18].

A. Non-Stationary Camera Video Completion Examples

In this part of our experiments, a fixed 8×8 size for the segmentation squares in the bandlet transform is used. The produced squares are considered as the blocks mentioned in the priority calculation (4) and patch matching (5). Each 3-D patch is defined over 3 consecutive frames having 3×3 blocks in each frame. In the motion estimation procedure, required to find the motion confidence (3), BMA's search range is 8 blocks along both of the x and y directions. Also, α is set to 0.5 for (8).

Fig. 8 illustrates the results of our algorithm on several 320×240 video sequences captured from digital camera produced videos, TV, and a video game. In each video sequence example, the top row and the bottom row includes the source frames and the completion resulted frames, respectively. The method is also applied on a classic video provided by the authors of [16]. The objective is reconstructing the frames damaged by considerably large coffee stain effect. The video completion result is illustrated in Fig. 9. In all the cases, the proposed method performs the completion task quite well.

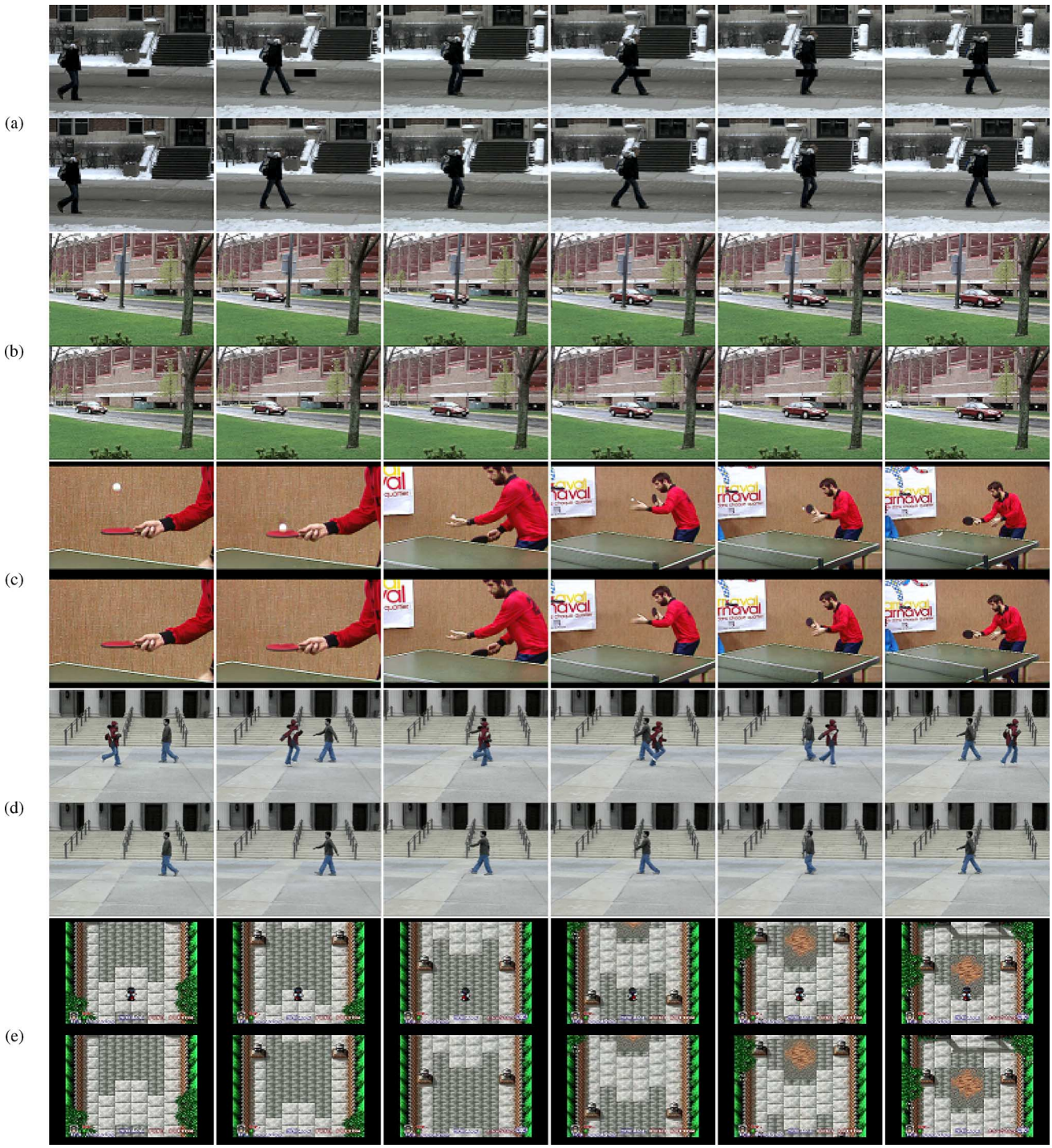


Fig. 8. Completion results of camera motion containing video sequences.

B. Stationary Camera Video Completion Examples

In these experiments, a fixed 16×16 size for the squares of the quadtree segmentation process in the bandlet transform is used for both of the background and partially occluded moving object completion tasks. In the latter completion, α is equal to 0.5 for the confidence value update.

Fig. 10(a) shows the completion task on a video sequence that includes a manually created occlusion, seen as a black rectangle on all the frames. The objective is to retrieve the occluded parts of the moving person, and also to reconstruct the background after the black rectangle removal. This is performed well using our method (Section III-B). In the background inpainting λ is initialized to 3 for soft thresholding. This is a reasonable value to reduce the number of iterations of Algorithm 2, as mentioned in Section III-B, and at the same time to obtain a good result considering that the background is not very textural and a degree of smoothness is acceptable in the final result. The occluded moving object completion algorithm is applied on another sequence, which consists of a moving person occluded by a stationary object as shown in Fig. 10(b).

Fig. 9. Completion results of Chaplin classic video. (a) Original frames. (b) Completed frames.

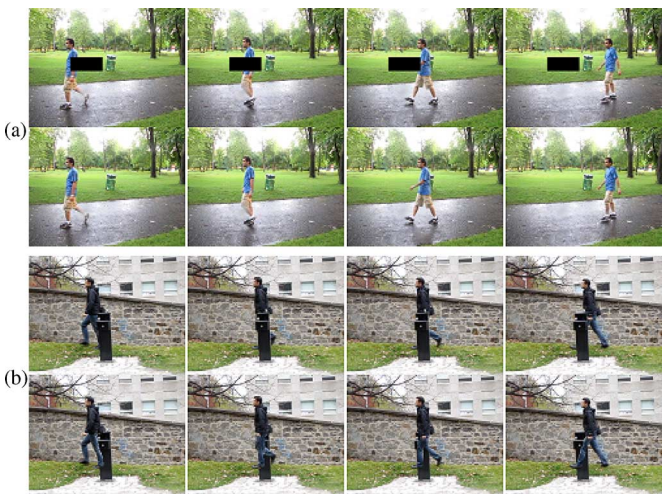


Fig. 10. Static camera captured videos completion.

C. Comparison Evaluation

The performances of video inapinting/completion methods are generally evaluated subjectively. An objective quality evaluation can be done only in presence of a ground-truth video. We use PSNR to evaluate the effectiveness of our method as it was used in [2] for still image inpainting evaluation. A manual damage is produced on an original video sequence. Then, the result of the completion method on the damaged video is compared with the original video sequence by computing the PSNR value for the corresponding frames of the original and the completion result video sequences. Fig. 11(a) shows a frame of the video chosen for evaluation which is damaged as in Fig. 11(b) and then completed as in Fig. 11(c). The green plot in Fig. 12 shows PSNR graph of all the 47 frames of the original video and the completion result sequence using 3-D patches in the method. It is worth noting that the value of PSNR is calculated only for the corresponding pixels of the damaged area and a smaller portion of the source pixels around them not the whole frame pixels. For almost all the frames, PSNR value is high, indicating visually pleasing completion results. The method is performed once again on the same sequence using 2-D patches instead of 3-D ones in order to give an insight how 3-D patch matching affects the performance of video completion. The resulting PSNR graph is shown in Fig. 12. The advantage of bandlet transform in video completion is evaluated by performing the completion task on the same video sequence, this time by using the traditional SSD in the patch matching. In this experiment SSD_B

Fig. 11. (a) Original frame. (b) Damaged frame. (c) Completion result (Frame number 20, PSNR $= 31.46$).

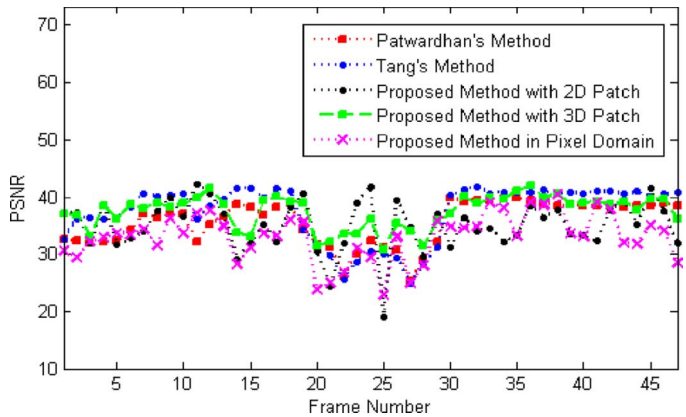


Fig. 12. Objective evaluation of the proposed video completion method. Average frame PSNR is 35.61 dB, 37.36 dB, 35.08 dB, 37.46 dB, and 33.07 dB for Patwardhan [12], Tang [18], proposed method with 2-D patches, proposed method with 3-D, and proposed method with 3-D patches only in pixel domain, respectively.

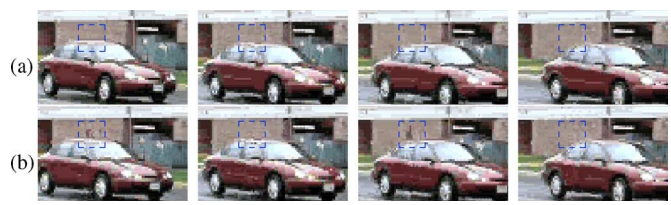


Fig. 13. From left to right, completion results of frames 12, 13, 14, and 15 of a sequence. (a) Our method. (b) Tang's method.

is set to 1 in (5) to see the effect of bandlets in the completion task. The completion results of our proposed 3-D patch-based method using bandlets jointly with the color information has a higher performance compared to the result of the same scheme performed only in the pixel domain, as shown in Fig. 12.

Two well-known video completion methods are the proposed methods in [12] and [18]. We did the same PSNR graph generation, i.e., computing PSNR for the completion results and the original sequence. The produced graphs are depicted in Fig. 12. The graphs and the computed average PSNR values of all the frames indicate a high performance for our proposed method compared to these two methods. This high performance is due to the effective role of bandlets in representing geometry features of the frames. Fig. 13 shows sample fames completed by our approach and Tang's method [18]. Fig. 13(b) shows a stationary region completed by different pixels in 4 consecutive frames leading to a flickering effect. This visual inconsistency is circumvented by our video completion method as illustrated in Fig. 13(a).

Undoubtedly, complexity is very important in video completion. Our method is based on the bandlet transform whose

efficient implementation is still a challenge. Therefore, in this paper, we focused mainly on the accuracy of our method.

V. Conclusions

In this paper, we made a distinction between stationary camera videos and non-stationary ones in a video completion task. The effectiveness of the bandlet transform in summarizing image geometry motivated our video completion methods. The method introduced for stationary scenes is a priority-based exemplar scheme. Priority of the missing region to be filled-in first is determined by reliable data and motion information in the bandlets structure. Patch matching procedure is also carried out in the bandlet domain. The proposed method for completion of videos captured by a static camera performs the filling-in process for background and foreground frames separately. For the background completion, inpainting is performed via an optimization in the bandlet domain. In the foreground completion process, a priority based method employing bandlets to determine each frame's structure is applied. Then, patch matching is used to find the most similar structure in the undamaged regions of the whole sequence. Finally, the completed foreground and background frames are combined to produce the final results. The experimental results indicate the considerable performance of our video completion method and the effectiveness of bandlets in this important task.

Acknowledgment

The authors would like to thank the anonymous reviewers and the Associate Editor for helpful and very insightful comments.

References

[1] J. Jia and C.-K. Tang, "Image repairing: Robust image synthesis by adaptive ND tensor voting," in *Proc. IEEE Computer Society Conf. Computer Vision and Pattern Recognition. (CVPR)*, Jun. 2003, pp. 643–650.

[2] O. Guleryuz, "Nonlinear approximation based image recovery using adaptive sparse reconstructions and iterated denoising—part I: Theory," *IEEE Trans. Image Process.*, vol. 15, no. 3, pp. 539–554, Mar. 2006.

[3] O. G. Guleryuz, "Nonlinear approximation based image recovery using adaptive sparse reconstructions and iterated denoising—part II: Adaptive algorithms," *IEEE Trans. Image Process.*, vol. 15, no. 3, pp. 555–571, Mar. 2006.

[4] M. Bertalmio, G. Sapiro, V. Caselles, and C. Ballester, "Image inpainting," in *Proc. 27th Annu. Conf. Computer Graphics and Interactive Techniques. (SIGGRAPH)*, 2000, pp. 417–424.

[5] T. F. Chan and J. Shen, "Mathematical models for local nontexture inpaintings," *SIAM J. Appl. Math*, vol. 62, pp. 1019–1043, 2001.

[6] M. Bertalmio, A. L. Bertozzi, and G. Sapiro, "Navier-stokes, fluid dynamics, and image and video inpainting," in *Proc. IEEE Computer Society Conf. Computer Vision and Pattern Recognition (CVPR)*, 2001, pp. I355–362.

[7] M. Bertalmio, "Strong-continuation, contrast-invariant inpainting with a third-order optimal pde," *IEEE Trans. Image Process.*, vol. 16, no. 7, pp. 1934–1938, Jul. 2006.

[8] M. Ghoniem, Y. Chahir, and A. Elmoataz, "Geometric and texture inpainting based on discrete regularization on graphs," in *Proc. 16th IEEE Int. Conf. Image Processing (ICIP)*, Nov. 7–10, 2009, pp. 1349–1352.

[9] O. Alatas, P. Yan, and M. Shah, "Spatio–temporal regularity flow (spref): Its estimation and applications," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 5, pp. 584–589, May 2007.

[10] A. Criminisi, P. Perez, and K. Toyama, "Region filling and object removal by exemplar-based image inpainting," *IEEE Trans. Image Process.*, vol. 13, no. 9, pp. 1200–1212, Sep. 2004.

[11] K. Patwardhan, G. Sapiro, and M. Bertalmio, "Video inpainting of occluding and occluded objects," in *Proc. IEEE Int. Conf. Image Processing (ICIP)*, Sep. 2005, pp. II69–72.

[12] K. A. Patwardhan, G. Sapiro, and M. Bertalmio, "Video inpainting under constrained camera motion," *IEEE Trans. Image Process.*, vol. 16, no. 2, pp. 4545–553, Feb. 2007.

[13] T. K. Shih, N. C. Tang, W.-S. Yeh, T.-J. Chen, and W. Lee, "Video inpainting and implant via diversified temporal continuations," in *Proc. 14th Annu. ACM Int. Conf. Multimedia*, Santa Barbara, CA, 2006, pp. 133–136.

[14] T. K. Shih, N. C. Tang, and J.-N. Hwang, "Exemplar-based video inpainting without ghost shadow artifacts by maintaining temporal continuity," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 19, no. 3, pp. 347–360, Mar. 2009.

[15] Y.-T. Jia, S.-M. Hu, and R. R. Martin, "Video completion using tracking and fragment merging," *Vis. Comput.*, vol. 21, pp. 601–610, Sep. 2005.

[16] Y. Wexler, E. Shechtman, and M. Irani, "Space-time completion of video," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 3, pp. 463–476, Mar. 2007.

[17] Y. Zhang, J. Xiao, and M. Shah, "Motion layer based object removal in videos," in *Proc. 7th IEEE Workshops Application of Computer Vision (WACV/MOTIONS'05)*, Jan. 2005, pp. 516–521.

[18] N. Tang, C.-T. Hsu, C.-W. Su, T. Shih, and H.-Y. Liao, "Video inpainting on digitized vintage films via maintaining spatiotemporal continuity," *IEEE Trans. Multimedia*, vol. 13, no. 4, pp. 602–614, Aug. 2011.

[19] S.-C. S. Cheung, J. Zhao, and M. V. Venkatesh, "Efficient object-based video inpainting," in *Proc. IEEE Int. Conf. Image Processing (ICIP)*, Atlanta, GA, Oct. 2006, pp. 705–708.

[20] J. Jia, Y.-W. Tai, T.-P. Wu, and C.-K. Tang, "Video repairing under variable illumination using cyclic motions," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 5, pp. 832–839, May 2006.

[21] C.-H. Ling, C.-W. Lin, C.-W. Su, Y.-S. Chen, and H.-Y. Liao, "Virtual contour guided video object inpainting using posture mapping and retrieval," *IEEE Trans. Multimedia*, vol. 13, no. 2, pp. 292–302, Apr. 2011.

[22] A. Mosleh, N. Bouguila, and A. B. Hamza, "A video completion method based on bandlet transform," in *Proc. IEEE Int. Conf. Multimedia and Expo. (ICME)*, Barcelona, Spain, Jul. 2011, pp. 1–6.

[23] A. Criminisi, G. Cross, A. Blake, and V. Kolmogorov, "Bilayer segmentation of live video," in *Proc. IEEE Computer Society Conf. Computer Vision and Pattern Recognition (CVPR)*, 2006, pp. 53–60.

[24] E. L. Pennec and S. Mallat, "Sparse geometric image representations with bandelets," *IEEE Trans. Image Process.*, vol. 14, no. 4, pp. 423–438, Apr. 2005.

[25] E. L. Pennec and S. Mallat, "Bandelet image approximation and compression," *SIAM Multiscale Model. Simul.*, vol. 4, pp. 992–1039, 2005.

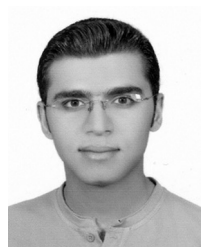
[26] S. Mallat and G. Peyre, "Surface compression with geometric bandelets," *ACM Trans. Graph.*, vol. 24, pp. 601–608, Jul. 2005.

[27] S. Mallat and G. Peyre, "Orthogonal bandlets bases for geometric image approximation," *Commun. Pure Appl. Math.*, vol. 61, pp. 1173–1212, Jul. 2008.

[28] S. Mallat and G. Peyre, "A review of bandlet methods for geometrical image representation," *Numer. Algorithms*, vol. 44, pp. 205–234, Mar. 2007.

[29] *Information Technology – Coding of Moving Pictures and Associated Audio for Digital Storage Media at up to About 1,5 Mbit/s*, ISO/IEC Std. 11, 1993.

[30] J. Starck, M. Elad, and D. Donoho, "Redundant multiscale transforms and their application for morphological component separation," *Adv. Imag. Electron Phys.*, vol. 132, pp. 287–348, 2004.



Ali Mosleh (S'11) received the B.S. and M.S. degrees in computer engineering from Iran Islamic Azad University, Maybod, Iran, and Science and Research Branch, Tehran, Iran, respectively. He is currently pursuing the Ph.D. degree in the Department of Electrical and Computer Engineering, Concordia University, Montréal, QC, Canada.

His research interests include computer vision, image and video processing, and multimedia coding.