Exemplar-based Video Completion with Geometry-guided Space-time Patch Blending

Maxime Daisy,* Pierre Buyssens, David Tschumperlé, Olivier Lézoray

(a) Video frame to inpaint.

(b) Exemplar-based inpainted frame by frame and close-up.

(c) Exemplar-based inpainted + our patch blending result, and close-up.

Figure 1: *Effect of our geometry-guided patch blending on a video completion result.*

Abstract

We propose an exemplar-based video completion algorithm together with a geometry-guided space-time artifact reduction technique. The proposed completion algorithm is the video extension of an inpainting algorithm proven to be effective on still images. Then, the proposed space-time artifact reduction technique blends multiple patches, guided by a tensor model in order to preserve local structures and textures as much as possible. The two contributions we propose are complementary, and provide video completion results of good quality without block-effect artifacts.

CR Categories: I.3.3 [Computer Graphics]: Picture/Image Generation— [I.3.4]: Computer Graphics—Graphics Utilities I.4.4 [Image Processing and Computer Vision]: Restoration— [I.4.9]: Image Processing and Computer Vision—Applications

Keywords: Space-time Patch Blending, Block-effect Artifact Reduction, Exemplar-based Video Inpainting.

1 Introduction and Context

Image and video inpainting allows to complete or restore images and videos continaing corrupted/missing data. Among the vast litterature on this topic [Guillemot and Le Meur 2014], the pattern-based methods use the self-similarity principle, and copy/paste known patches to the unknown area to restore the image/video.

A major difficulty of these methods is to avoid visible seams between reconstructed patches. While [Wexler et al. 2007; Newson et al. 2014] propose a multiresolution scheme to inpaint images and videos, and to use multiple patches for the reconstruction. The major flaw of such a methods is that it tends to blur the reconstructed textures.

In this paper, we first go back to the seminal single-scale greedy method of [Criminisi et al. 2004] and adapt it to the video inpainting with the integration of many improvements proposed in [Buyssens et al. 2015]. In a second time, we propose an anisotropic space-time blending algorithm that considerably reduces typical block-effect artifacts reminiscent of [Criminisi et al. 2004] while keeping sharp structures and textures.

2 Exemplar-based Video Inpainting

We propose an adaptation of the exemplar-based image inpainting algorithm [Criminisi et al. 2004] to video data.

First, we change the shape of the patch to use. As the temporal dimension does not have the same meaning than the spatial one, we do not use cubic patches (same dimensions in space and time), but parallelepipoidic patches. The main reason is that space-time patches capture the local instant motion in a video, and we do not want to propagate this motion too much. This allows to avoid temporal inconsistencies as much as possible.

For the reconstruction, we propose to adapt the search scheme described in [Buyssens et al. 2015] for videos. For a patch to be reconstructed at point p_n, this method extracts all the best match locations $\Phi = \{\hat{p}_0, \hat{p}_1, \ldots, \hat{p}_m\}$ of reconstructed patch locations $\{p_1, p_2, \ldots, p_m\}$ inside a window W centered at p_n and of a user-defined size s_W. The latter is generally provided as a factor of the patch size. The $\hat{p}_i$ are then used for finding search sites $q_i = \hat{p}_i + p_n - p_i$ to perform a window lookup for each window $w_i \in \{w_1, w_2, \ldots, w_m\}$. The size of these windows is $s_{w_i} = \alpha s_W \sqrt{|\Phi|}$, where α is a parameter controlling the amount of space to give to the small windows w.r.t. the initial window search size. This allows to keep a similar complexity of the patch lookup whatever the number of sites to be sought. The idea behind this scheme is to reconstruct large portions of video with highly correlated patchs chunks. This method, parent of those in [Ashikhmin 2001; Barnes et al. 2009], enables an efficient lookup in term of time and reconstruction quality.

*This research was supported by French national grant *Action 3DS*.

Using these adaptations, the proposed video inpainting technique provides good results in term of geometry and texture reconstruction. However, one can notice that it suffers from block-effect artifacts along the space and the time dimensions (see Fig. 1, middle).

3 Space-time Patch Blending

In this section we propose an algorithm that creates a smarter transition between reconstructed patches within space-time greedy inpainting results $I : p \in \mathcal{I} \mapsto I(p) \in \mathbb{R}$ with $\mathcal{I} \subset \mathbb{N}^{3+}$. We define a geometric tensor model that allows keeping well the image structures and textures while creating these transitions.

Our *patch blending algorithm* mixes the overlapping data of multiple patches in greedy patch-based inpainting results (see Fig. 2) in order to remove the seams between the reconstructed patch chunks. This process mainly contains the following steps:
1) Computation of our blending tensor model: further described in the Section 4, the tensor field $\mathbf{B}(p)$ is computed on the masked video and reconstructed it from the inpainting correspondence map.
2) Model regularization: with the same idea of anisotropic image smoothing [Tschumperle and Deriche 2005], the tensors extracted from the image are regularized by an isotropic smoothing process.
3) Spatial blending of the inpainted video: we compute the final video J, using $\mathbf{B}(p)$, by applying the following formula (see Fig. 2 for notation explanations):

$$J(p) = \frac{\sum_{i \in \{1,\dots,|\Psi_p|\}} w_{\mathbf{B}}(p, p_i)\, \psi_{\hat{p}_i}(p - p_i)}{\varepsilon + \sum_{i \in \{1,\dots,|\Psi_p|\}} w_{\mathbf{B}}(p, p_i)} \tag{1}$$

where Ψ_p is the set of the centers of the reconstructed patches con-

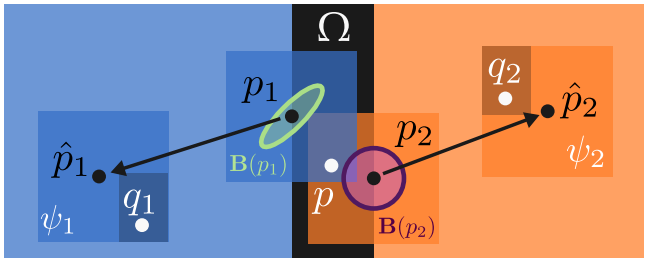


Figure 2: *Principle of our patch blending algorithm: one wants to blend at p using source patches ψ_1 and ψ_2 with the blending tensors $\mathbf{B}(p_1)$ and $\mathbf{B}(p_2)$. While an isotropic blending is applied at p_2, an anisotropic one is applied at p_1.*

taining p, $w_{\mathbf{B}}(p, q) = e^{-X^T \mathbf{B}(p)^{-1} X}$ is an anisotropic Gaussian weight function with $X = q - p$ and $\mathbf{B}(p)$ is our blending tensor model. Before going to the details of the latter (Section 4), we do a high-level description of their spatial and temporal behavior.

Spatial behavior: the patch-based blending model must be able to cope with different spatial components:
• In case of *flat areas* for a frame $I^{(t)}$, the blending must be of high amplitude in any directions to flatten small seams.
• In case of a *textured area* of frame $I^{(t)}$, the blending has to be applied in any directions, with a moderate amplitude such that it does not degrade too much the texture pattern.
• In case of a *structure component*, the blending has to follow the structure direction with a high amplitude such that its sharpness is preserved.

Temporal behavior: while dealing with the spatial components, the blending model is able to cope with different temporal cases:
• In case of *no or small instant motions*, the inpainting result has to be blended with a high amplitude along the time dimension to flatten temporal seams reminiscent of the inpainting.
• In case of *moderate to high amplitude motions*, the blending along the time dimension has to be small or null to avoid blending an object at the frame t with another at the frame $t + 1$.

4 Tensor Model for Space-time Blending

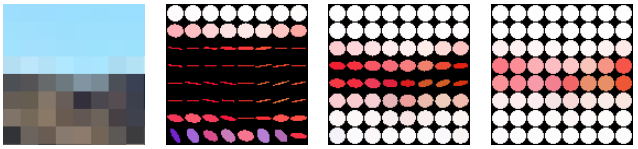
To handle all the spatial and temporal video configurations described above, we define a unified blending model. This model must be able to tackle flat areas with its isotropic properties, but also to be aware of the video structures using anisotropy. Therefore, we chose a tensor model that is the lightest model able to describe both isotropy and anisotropy. The model we propose is defined as:

$$\mathbf{B} = \sum_{j \in \{1,2,3\}} \lambda_{\mathbf{B}j} \mathbf{e}_{\mathbf{B}j} {\mathbf{e}_{\mathbf{B}j}}^T = \underbrace{\sum_{i \in \{1,2\}} \lambda_{\mathbf{B}i} \mathbf{e}_{\mathbf{B}i} {\mathbf{e}_{\mathbf{B}i}}^T}_{\text{spatial term}} + \underbrace{\lambda_{\mathbf{B}t} \mathbf{e}_{\mathbf{B}t} {\mathbf{e}_{\mathbf{B}t}}^T}_{\text{temporal term}} \tag{2}$$

where eigenvectors $\mathbf{e}_{\mathbf{B}i}$ represent the preferred orientations for the blending, and eigenvalues $\lambda_{\mathbf{B}i}$ $\lambda_{\mathbf{B}t}$ represent the blending bandwidths in space and time respectively. In order to keep the control of the overall anisotropy of $\mathbf{B}$, we propose the following definition inspired by partial differential equations for diffusion [Tschumperle and Deriche 2005] for its eigenvalues:

$$\lambda_{\mathbf{B}i} = \frac{\sigma}{(1 + \sum_{j \in \{1,2,3\}} \hat{\lambda}_{\mathbf{B}j})^{\gamma_i}} \quad \text{and} \quad \lambda_{\mathbf{B}t} = \frac{\sigma_t}{(1 + \sum_{j \in \{1,2,3\}} \hat{\lambda}_{\mathbf{B}j})^{\gamma_t}} \tag{3}$$

where σ and σ_t are the user-defined spatial and temporal blending



(a) Image. (b) $\frac{\gamma_2}{\gamma_1} \mapsto \infty$ (c) $\frac{\gamma_2}{\gamma_1} > 1$ (d) $\frac{\gamma_2}{\gamma_1} \approx 1$

Figure 3: *Illustration of the anisotropy control of the blending tensors in a 2D image through the parameters γ_i.*

bandwidth respectively. As more described further, $\gamma_i, i \in \{1, 2\}$ (resp. γ_t) control the anisotropy of $\mathbf{B}$ in the spatial (resp. temporal) dimension. The $\hat{\lambda}_{\mathbf{B}j}, j \in \{1, 2, 3\}$ are normalized eigenvalues that depend on the local video geometry and are computed as follows:

$$\hat{\lambda}_{\mathbf{B}i} = \frac{\lambda_{\mathbf{S}i}}{\max_{\mathcal{I}} \lambda_{\mathbf{S}i}} \tag{4}$$

$\lambda_{\mathbf{S}i}, i \in \{1, 2, 3\}$ are the eigenvalues of the structure tensor $\mathbf{S}$, that gives a description of local geometry of color images. This normalization step aims at making the $\hat{\lambda}_{\mathbf{B}i}$ to be independent of the image value range. This is not the case with structure tensors [Di Zenzo 1986] defined as follows:

$$\mathbf{S} = \sum_{k \in \{R,G,B\}} \overrightarrow{\nabla I_k} \,.\, \overrightarrow{\nabla I_k}^T \tag{5}$$

Originally defined for 2-dimensional images, structure tensors can be adapted to video data by adding the local temporal derivative of the video such that $\overrightarrow{\nabla I_k} = \begin{pmatrix} \frac{\partial I_k}{\partial x} & \frac{\partial I_k}{\partial y} & \frac{\partial I_k}{\partial t} \end{pmatrix}^T$. There are different possibilities to compute the temporal derivative $\frac{\partial I}{\partial t}$ like

the optical flow for example. In our experiments, for a frame $I^{(t)}$ of the video I, we compute the gradient of the pixel values: $\frac{\partial I^{(t)}}{\partial t} = \frac{I^{(t+1)} - I^{(t-1)}}{2}$.

As a large variety of images exists, it is good to have the control on the way that one wants to locally apply the blending. Therefore, we chose to introduce the parameters $\gamma_i, i \in \{1, 2\}$ ($\gamma_1 < \gamma_2$) and γ_t, that aim at changing the overall shape of the blending tensors. For an image that has strong contours to preserve, we choose $\gamma_2 >> \gamma_1$, e.g. $\gamma_2 = 15, \gamma_1 = 0.5$ (see Fig. 3(b)). On the other hand, for a low-contrast image we choose $\gamma_2 \approx \gamma_1$, e.g., $\gamma_2 = \gamma_1 = 0.5$ (see Fig. 3(d)) to blend in all directions. Fig. 3 shows different configurations of γ_i for the spatial plane xy, and their effect on the blending tensors. Note that by using $\gamma_1 = \gamma_2$, one is able to apply *isotropic* spatial patch blending. The blending tensor

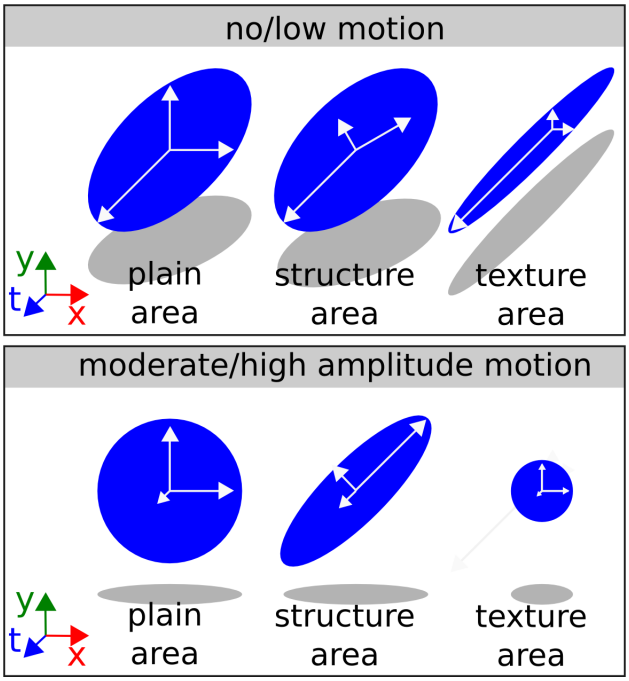


Figure 4: *Space-time blending configuration depending on the local video structure and motion.*

model we have defined follows the local properties summarized in Fig. 4. An example of blending tensors is given Fig. 5 and exhibits the projections of the blending tensors along the xz plane ((a) and (b)), xy plane ((c) and (d)). One can see that tensors are aligned with the motion direction in (b), and with the objects contours in (d). These properties allow the patch blending to be respectful of the structures and the textures.

5 Results and Conclusions

In Fig. 6 we compare the results of our method to those without blending, and those of [Newson et al. 2014]. At first, one can notice that from exemplar-based inpainting results without blending (second row) to our results (last row), the block-effect artifacts are removed while the structures of the objects remain intact. Also, our method provides results with a similar visual quality of those of [Newson et al. 2014].

In this paper we presented an exemplar-based video inpainting and a space-time patch blending technique that is able to reduce space-time block-effect artifacts. Together, this contributions are able to

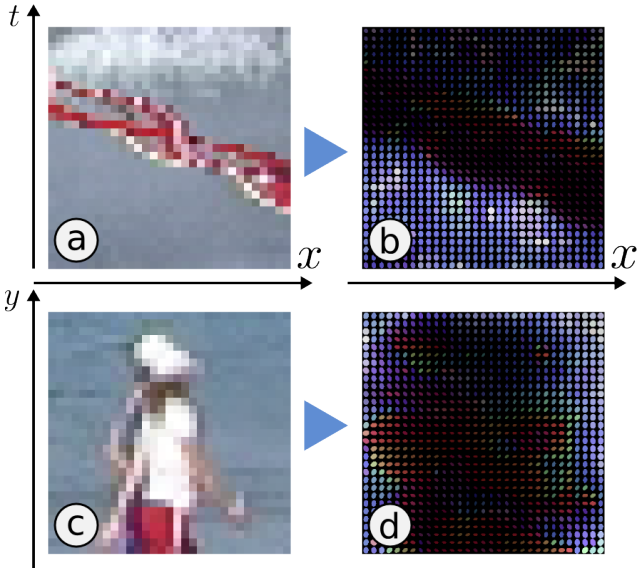


Figure 5: *Space-time blending tensor fields projection on xy and xt planes (right) computed on sample images (left). The women are walking from the left to the right in the video.*

produce good quality video inpainting result without altering the input video content. In future works, we will use an optical flow estimation to enhance the accuracy of our geometry model for video patch blending.

References

ARIAS, P., FACCIOLO, G., CASELLES, V., AND SAPIRO, G. 2011. A variational framework for exemplar-based image inpainting. *Int. J. Comput. Vision 93*, 3 (July), 319–347.

ARSIGNY, V., FILLARD, P., PENNEC, X., AND AYACHE, N. 2005. Fast and simple calculus on tensors in the log-euclidean framework. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2005*. Springer, 115–122.

ASHIKHMIN, M. 2001. Synthesizing natural textures. In *Interactive 3D graphics*, ACM, 217–226.

BARNES, C., SHECHTMAN, E., FINKELSTEIN, A., AND GOLDMAN, D. B. 2009. PatchMatch: A randomized correspondence algorithm for structural image editing. *ACM T. Graphic. 28*, 3 (Aug.).

BUYSSENS, P., DAISY, M., TSCHUMPERLE, D., AND LÉZORAY, O. 2015. Exemplar-based inpainting: Technical review and new heuristics for better geometric reconstructions. *Image Processing, IEEE Transactions on 24*, 6, 1809–1824.

CRIMINISI, A., PÉREZ, P., AND TOYAMA, K. 2004. Region filling and object removal by exemplar-based image inpainting. *IEEE Transactions on Image Processing 13*, 9 (Sept.), 1200–1212.

DI ZENZO, S. 1986. A note on the gradient of a multi-image. *Computer Vision, Graphics, and Image Processing 33*, 1, 116 – 125.

EFROS, A., AND LEUNG, T. K. 1999. Texture synthesis by non-parametric sampling. In *International Conference on Computer Vision*, vol. 2, IEEE, 1033–1038.

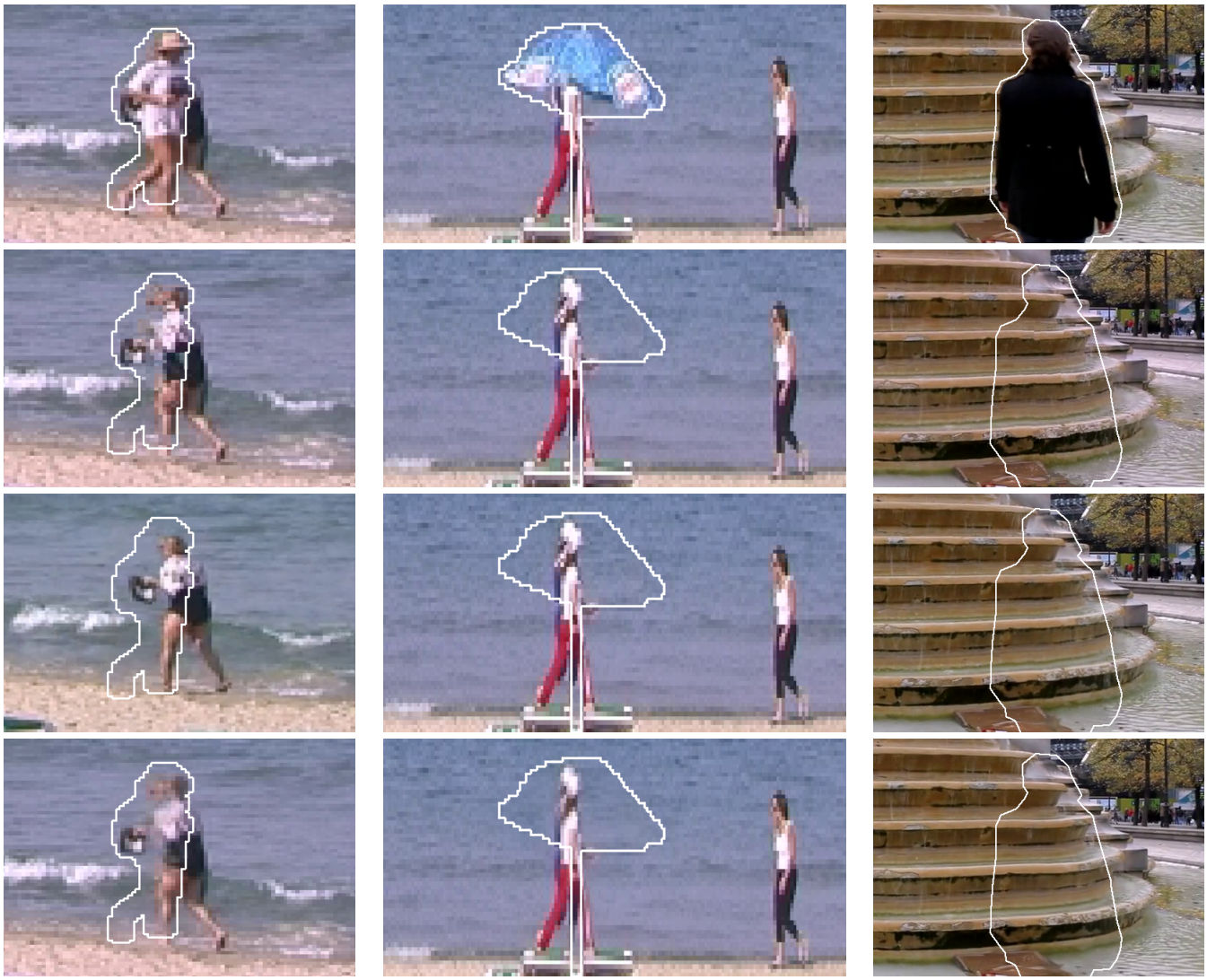


Figure 6: *Comparison of our method (fourth row) to the result without blending (second row), and to those obtained with [Newson et al. 2014] (thrid row). The first row is the original video frame.*

GUILLEMOT, C., AND LE MEUR, O. 2014. Image inpainting: Overview and recent advances. *IEEE Signal Processing Magazine 31*, 1, 127–144.

KROTKOV, E. P. 1989. *Active computer vision by cooperative focus and stereo*. Springer-Verlag New York, Inc.

LE MEUR, O., GAUTIER, J., AND GUILLEMOT, C. 2011. Examplar-based inpainting based on local geometry. In *ICIP*, 3401–3404.

LE MEUR, O., EBDELLI, M., AND GUILLEMOT, C. 2013. Hierarchical super-resolution-based inpainting. *IEEE Transactions on Image Processing 22*, 10, 3779–3790.

MASNOU, S., AND MOREL, J.-M. 1998. Level lines based disocclusion. In *International Conference on Image Processing*, 259–263.

NEWSON, A., ALMANSA, A., FRADET, M., GOUSSEAU, Y., PÉREZ, P., ET AL. 2014. Video inpainting of complex scenes. In *SIAM Journal on Imaging Sciences*, vol. 7:4, 1993–2019.

PATWARDHAN, K. A., SAPIRO, G., AND BERTALMIO, M. 2005. Video inpainting of occluding and occluded objects. In *IEEE International Conference on Image Processing*, vol. 2, IEEE, II–69.

PÉREZ, P., GANGNET, M., AND BLAKE, A. 2003. Poisson image editing. *ACM Transactions on Graphics 22*, 3 (July), 313–318.

TSCHUMPERLE, D., AND DERICHE, R. 2005. Vector-valued image regularization with pdes: A common framework for different applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence 27*, 4, 506–517.

WEXLER, Y., SHECHTMAN, E., AND IRANI, M. 2007. Space-time completion of video. *IEEE Trans. Pattern Anal. Mach. Intell. 29*, 3 (Mar.), 463–476.