

# Content-Based Projections for Panoramic Images and Videos

Leonardo K. Sacht, Paulo C. Carvalho, Luiz Velho  
*Vision and Graphics Laboratory - Visgraf*  
*Instituto Nacional de Matematica Pura e Aplicada - IMPA*  
*Rio de Janeiro, Brazil*  
*Email: {leo-ks, pcezar, lvelho}@impa.br*

**Abstract**—This paper<sup>1</sup> studies the problem of obtaining panoramic images, i.e., images of wide fields of view. We start by analyzing previous approaches, in order to understand the difficulties of this problem. Then we discuss in detail the work by Carroll et al., entitled “Optimizing content-preserving projections for wide-angle images” and show results of this method that prove that it produces good results in a variety of scenes. We discuss some aspects that were not emphasized in that work, such as the optimization method used to produce the final result.

In addition, we propose some important extensions: an interface with some additional features to the one proposed by them; and Computer Vision methods to detect faces and straight lines in equirectangular images, which are very important features for the method.

We also provide a novel study about panoramic videos, i.e., videos where each frame is constructed from a wide FOV. We introduce a mathematical model for this problem, discuss desirable temporal coherence properties, formulate equations that represent these properties, propose an optimization solution for a particular case and point future directions.

**Keywords**—Viewing sphere, panoramic images, panoramic videos.

## I. INTRODUCTION

In this section, we motivate the panoramic image problem, state the necessary terminology and briefly discuss previous approaches.

### A. Motivation

One of the main motivations for the panoramic image problem is that common cameras capture just a limited field of view (FOV), usually up to 90 degrees, while our eyes see a wider FOV with no obvious distortions. The panoramic images can be used to extrapolate our perception, since they can capture FOVs beyond the human eye. Also, a panoramic image allows us to better represent an entire scene, since there may be important parts in a scene that could not be seen under a limited FOV.

### B. The Viewing Sphere

In this work, any scene observed from a fixed viewpoint at a given moment will be modeled as the unit sphere centered

at the viewpoint ( $\mathbb{S}^2 = \{(x, y, z) \in \mathbb{R}^3 | x^2 + y^2 + z^2 = 1\}$ ) on which each point has an associated color, the color that is seen when one looks toward this point. We will refer to it as the *viewing sphere*. Consider the longitude/latitude representation of the sphere  $\mathbf{r}$ :

$$\begin{aligned} \mathbf{r} : [-\pi, \pi] \times [-\frac{\pi}{2}, \frac{\pi}{2}] &\rightarrow \mathbb{S}^2 \\ (\lambda, \phi) &\mapsto (\cos(\lambda) \cos(\phi), \sin(\lambda) \cos(\phi), \sin(\phi)) \end{aligned}$$

With this correspondence, we can represent viewing spheres as images which we call *equirectangular images*. In figure 1 we show an example of equirectangular image.



Figure 1. Each point in the image represents a point on the  $(\lambda, \phi)$  domain. Image title: “San Marco Plaza”, by Flickr user Veneboer, taken from [1].

### C. Problem Statement

With the notations of the last section we formulate the panoramic image problem as that of finding a projection

$$\begin{aligned} \mathbf{u} : S \subseteq \mathbb{S}^2 &\rightarrow \mathbb{R}^2 \\ (\lambda, \phi) &\mapsto (u(\lambda, \phi), v(\lambda, \phi)) \end{aligned}$$

with desirable properties.

There are some known examples of these projections: the perspective projection has the desirable property of mapping straight lines in the scene to straight lines in the final result, but it stretches objects too much when the FOV increases. Mercator and stereographic projections preserve objects well, but bend lines. For more details about these standard projections, we refer the reader to [2], pp. 10-20.

<sup>1</sup>This is a short version of the Master Thesis of the first author. The full text and additional material can be found at: [http://w3.impa.br/~leo-ks/msc\\_thesis](http://w3.impa.br/~leo-ks/msc_thesis).

#### D. Previous Approaches

One of the first works that considered the panoramic image problem was the one by Zorin et al. [3]. In this work, the authors formalized perceptual properties that a wide-angle image should have and showed that two important properties could not be satisfied simultaneously: the preservation of object shapes and of straight lines. Then they used an optimization framework to obtain a family of projections that depend on a parameter  $\lambda \in [0, 1]$ . When  $\lambda = 0$  all the shapes are preserved, when  $\lambda = 1$  all the straight lines are mapped to straight lines in the result and intermediate values represent the tradeoff between the two properties. We show in figure 2 a result produced with  $\lambda = \frac{1}{2}$ .



Figure 2. Result produced by the method described in [3]. FOV: 150 degree longitude/150 degree latitude.

In [4] local projections for the objects in the scene are adopted to correct distortions caused by projections such as the perspective projection. Therefore, the projection is obtained based on the scene content.

Other approach that used the scene content to obtain the final projection was [5]. The authors used local projections for the objects but also multiple perspective projections for the background in a way that fits the geometry of the scene. A result by this method is shown in figure 3.



Figure 3. Result produced by the method described in [5]. FOV:180/90. Unpleasant orientation discontinuities appear on the ceiling.

## II. OPTIMIZING CONTENT-PRESERVING PROJECTIONS FOR WIDE-ANGLE IMAGES

As we saw in the previous approaches, the main difficulty in producing panoramic images is to conciliate preservation of straight lines and object shapes. Other properties such as smoothness of the projection, dependence on the scene content and formalization of the distortions also proved to be important.

The work by Carroll et. al ([6]) has all these properties and produce good results in a variety of scenes and for arbitrarily wide FOVs. They model undesirable distortions in panoramic images, formulate energies that measure how a panoramic image contains these distortions and obtain the least distorted panoramic image via optimization. This section is devoted to analyze this approach, presenting details that were not considered in [6].

We impose a uniform discretization ( $\{\lambda_{ij}, \phi_{ij}\}_{i,j}$ ) of the longitude/latitude domain and look for the projection on these vertices. The final projection is obtained by bilinear interpolation of the positions  $\{(u(\lambda_{ij}, \phi_{ij}), v(\lambda_{ij}, \phi_{ij}))\}_{i,j}$ .

#### A. Conformality

We use the geometric concept of conformal mappings to model preservation of object shapes. This concept is appropriate to model this property because conformal mappings are locally a rotation and/or a scaling.

In the context of panoramic images (mappings  $\mathbf{u} : \mathbb{S}^2 \rightarrow \mathbb{R}^2$ ), a projection is conformal if and only if it satisfies the Cauchy-Riemann equations:

$$\frac{\partial u}{\partial \phi} = -\frac{1}{\cos(\phi)} \frac{\partial v}{\partial \lambda} \text{ and } \frac{\partial v}{\partial \phi} = \frac{1}{\cos(\phi)} \frac{\partial u}{\partial \lambda}.$$

The proof of this fact is in [2], p. 39.

By approximating the partial derivatives by finite differences we obtain an energy term that measures how much a discretized projection deviates from conformality:

$$E_c = \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} w_{ij}^2 \left( \frac{v_{i,j+1} - v_{ij}}{\Delta \lambda} + \cos \phi_{ij} \frac{u_{i+1,j} - u_{ij}}{\Delta \phi} \right)^2 + \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} w_{ij}^2 \left( \frac{u_{i,j+1} - u_{ij}}{\Delta \lambda} - \cos \phi_{ij} \frac{v_{i+1,j} - v_{ij}}{\Delta \phi} \right)^2$$

Above,  $w_{ij}$  are spatially varying weights (section II-D),  $u_{ij} = u(\lambda_{ij}, \phi_{ij})$  and  $v_{ij} = v(\lambda_{ij}, \phi_{ij})$ .

By defining a vector  $\mathbf{x}$  with the unknowns  $u_{ij}$  and  $v_{ij}$ ,  $E_c$  can be rewritten as  $E_c = \|\mathbf{C}\mathbf{x}\|^2$ , where  $\mathbf{C}$  is a matrix.

#### B. Straight lines

Instead of making straight all possible lines in the scene (such as the perspective projection), the method uses the information provided by the user in the interface (section III-C) to straighten only the lines specified by her. An output of the interface is shown in figure 4.



Figure 4. The user specifies lines that she wants to be straight in the final result. She also can specify orientation of the lines in the final result: red stands for vertical lines and blue for horizontal ones. Green lines have no specified orientation.

For horizontal and vertical lines we formulate an energy (details in [2], pp.44-51) that measures how much the specified lines deviate from being straight and in the specified orientation. This energy can be written in the matrix form

$$E_{lo} = \|LO\mathbf{x}\|^2,$$

where  $LO$  is a matrix.

For lines with no specified orientation, we have to alternate between minimizing two different energies:

$$E_{lo} = \|LO\mathbf{A}\mathbf{x}\|^2 \text{ and } E_{ld} = \|LDA\mathbf{x}\|^2,$$

where  $LOA$  and  $LDA$  are matrices. The result of minimizing one energy is used to construct the other energy. This alternation is necessary to avoid nonlinearities inside the quadratic terms (all the details are in [2], pp.44-51).

### C. Smoothness

Minimizing only conformality and straight line energies may lead to unpleasant artifacts, as shown in figure 5 (left). These artifacts occur because the projection has to change too much near the line segments to satisfy the line constraints.

We formulate an energy (the details can be found in [2], pp.56-58) that measures the variation of the differential north vector  $\mathbf{h} = \begin{pmatrix} \frac{\partial u}{\partial \phi} & \frac{\partial v}{\partial \phi} \end{pmatrix}^T$  along the projection. The variation of  $\mathbf{h}$  is given by

$$\frac{\partial \mathbf{h}}{\partial(\lambda, \phi)} = \begin{pmatrix} \frac{\partial \mathbf{h}}{\partial \lambda} & \frac{\partial \mathbf{h}}{\partial \phi} \end{pmatrix} = \begin{pmatrix} \frac{\partial^2 u}{\partial \phi \partial \lambda} & \frac{\partial^2 u}{\partial \phi^2} \\ \frac{\partial^2 v}{\partial \phi \partial \lambda} & \frac{\partial^2 v}{\partial \phi^2} \end{pmatrix}.$$

Thus, ideally, we should have the four derivatives above equal to zero. By discretizing them using finite differences and multiplying them by the spatially varying weights  $w_{ij}$ , we obtain an energy that has the form  $E_s = \|S\mathbf{x}\|^2$ , where  $S$  is a matrix.

The result obtained minimizing this energy together with the previous ones is shown in figure 5 (right).

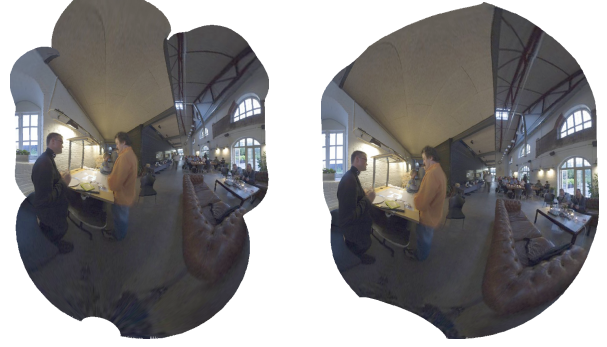


Figure 5. Left: Result without the smoothness energy. Right: Result with the smoothness energy (final result of the method). FOV: 285/180.

### D. Spatially-varying weights

For each vertex of the discretization, we define weights  $w_{ij}$  that control the strength of conformality and smoothness energies in different regions of the projection. A complete discussion about the influence of these weights is provided in [2], pp.58-62. They are given by  $w_{ij} = 2w_{ij}^L + 2w_{ij}^S + 4w_{ij}^F + 1$ , where  $w_{ij}^L$  has higher values near line endpoints,  $w_{ij}^S$  has higher values in more salient regions and  $w_{ij}^F$  has higher values near faces. The most effective weights are the face weights and we dedicate section III-A to explain the method we developed to detect faces in equirectangular images.

### E. Total energy and minimization

In order to take into account all the energies we just presented, we formulate energies that are sums of the previous ones. We alternate between minimizing  $E_d$  and  $E_o$

$$E_d = w_c^2 E_c + w_s^2 E_s + w_l^2 \sum_{l \in L_f} E_{lo} + w_l^2 \sum_{l \in L \setminus L_f} E_{ld}, \text{ and}$$

$$E_o = w_c^2 E_c + w_s^2 E_s + w_l^2 \sum_{l \in L_f} E_{lo} + w_l^2 \sum_{l \in L \setminus L_f} E_{ld},$$

where  $L = \{\text{lines marked by the user}\}$ ,  $L_f = \{\text{lines with fixed orientation}\}$ ,  $w_c = 0.4$ ,  $w_s = 0.05$ ,  $w_l = 1000$ . This alternation is necessary due to the green lines. Usually we minimize three times each energy to get visual convergence.

Using the matrices previously obtained, we can put both energies in the matrix form  $E_d = \|A_d\mathbf{x}\|^2$ ,  $E_o = \|A_o\mathbf{x}\|^2$ . Thus the problem becomes minimizing an energy of the form  $E(\mathbf{x}) = \|A\mathbf{x}\|^2$  at each iteration, which leads to finding the eigenvector of  $A^T A$  associated to the third smallest eigenvalue, as shown in [2], pp.63-65. Since this task is impractical due to the size of the problem (thousands of variables), we replace  $E$  by

$$\tilde{E}(\mathbf{x}) = E(\mathbf{x}) + \varepsilon \|\mathbf{x} - \mathbf{y}\|^2 = \|A\mathbf{x}\|^2 + \varepsilon \|\mathbf{x} - \mathbf{y}\|^2,$$

where  $\varepsilon = 10^{-6}$  and  $\mathbf{y}$  is the stereographic projection. As shown in [2], pp.65-67, the global minimizer of  $\tilde{E}$  is  $\hat{\mathbf{x}} = (A^T A + \varepsilon I)^{-1}(\varepsilon \mathbf{y})$ , what leads us to replace an eigenvalue problem by solving a sparse and symmetric linear system.

## F. Results

In this section we show results produced by the method explained in the last section. We used about 40,000 vertices to discretize the viewing sphere and the time of computation was about one minute to produce each example.

The lines were marked in the interface we explain in section III-C and we used Matlab to solve the linear systems.

Figure 6 shows an example where many lines were marked (sixty-nine) and cover a good part of the scene.

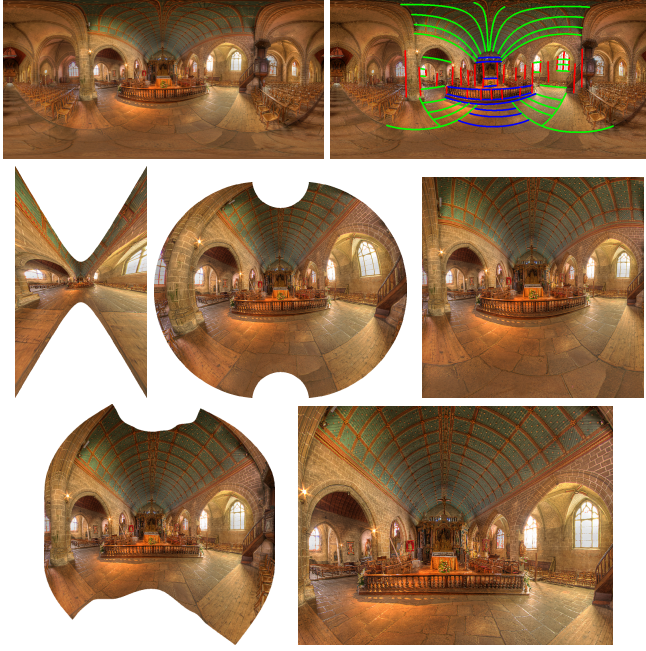


Figure 6. Top: Input image and marked lines. Middle: Perspective, Stereographic and Mercator projections. Bottom: Result produced by the method and cropped result. FOV: 210/140.

Figure 7 shows that the method produces good results for very wide FOVs.



Figure 7. Top: Input Image and marked lines. Bottom: Result produced by the method (cropped). FOV: 360/170.

Figure 8 shows a result where faces were detected. This detection is important to control distortions in face regions.



Figure 8. Top: Marked lines and detected faces. Bottom: Result produced by the method and cropped result. FOV: 150/140.

In figure 9, we show a result to be compared with the one produced by the method in [5] (see figure 3). We notice that the unpleasant discontinuities are not present in this result.

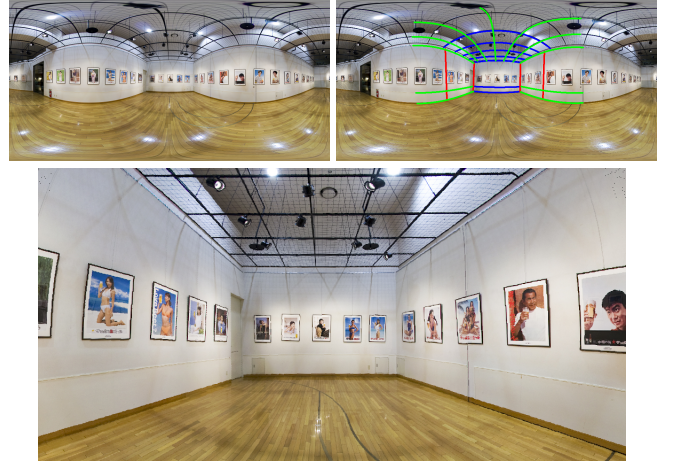


Figure 9. Top: Input image and marked lines. Bottom: Result produced by the method (cropped). FOV: 180/100.

The results just presented show that the method produce good results in a variety of situations. In [2], pp.73-96, more results, details and comparisons with other projections are provided. We also show that failure cases may happen sometimes, specially if the user forgets to mark the most important lines in the scene.

### III. EXTENSIONS

Besides carrying out a conclusive analysis of [6], exposing many details that were not considered in this reference (optimization methods, for instance), we also propose extensions that are important for the method.

#### A. Face detection in equirectangular images

The authors of [6] suggested steps to detect faces and we used their ideas to develop a method to detect faces in equirectangular images which is independent of the method to produce panoramic images, but can be integrated to it by using the face information to determine the weights  $w_{i,j}^F$ .

The key idea of the method is that faces can be very distorted in the equirectangular image, especially near the poles. In this case, a standard method to detect faces (we used [7]) might not detect them. Since the Mercator projection is conformal, the faces are well preserved in a Mercator image. Thus, the method consists of detecting faces in the Mercator image and mapping them back to the equirectangular domain (for details, see [2], pp.128-137). A result of this method is shown in figure 8 (top/right).

#### B. Line detection in equirectangular images

The interface suggested in [6] asks the user to specify all the important lines in the scene, by clicking on their endpoints (figure 4 shows a result of this process). This task may be sometimes tedious and/or the result may be imprecise. We developed a method to help the user in this task. The key idea of our method is that the perspective projection preserves lines. We project the sphere using six different perspective projections that cover the entire sphere. Then the Hough transform is used to detect lines in each perspective image and the information is mapped back to the equirectangular image. For details, see [2], pp.137-150. A result of our method is shown in figure 10.

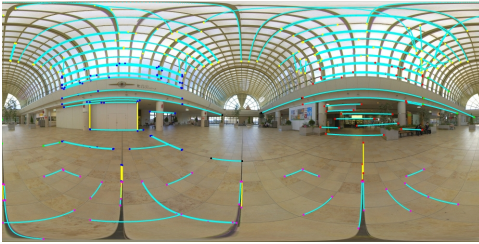


Figure 10. 161 line segments detected.

#### C. User Interface

Besides giving the possibility of specifying lines and their orientations, as in [6], our interface allows the user to specify what FOV will be projected, number of vertices of the sphere and number of iterations of the method. In ([8]), one can find a video of our interface working and in [2], pp. 118-122, one can find a manual of how to use it. The application itself is available upon request.

### IV. PANORAMIC VIDEOS

In this section, we consider the problem of obtaining panoramic videos, i.e., videos where each frame is a panoramic image. This problem has additional motivations to the image problem, such as movie industry and sport broadcasting applications. As far as we know, this problem has not been considered yet in the literature.

We start by separating the problem in three cases, in order to make it easier to solve each case. Then we state desirable properties that a panoramic video should have and mathematically model the problem. We finish by presenting a solution for case 1.

#### A. The three cases

Our separation is based on the position from where the scene is being filmed (Viewpoint-VP), the FOV that is being projected and the objects in the scene. The three cases are:

- Case 1: Stationary VP, stationary FOV, moving objects;
- Case 2: Stationary VP, moving FOV, stationary objects;
- Case 3: Moving VP.

#### B. Desirable properties

In analogy to the image problem, we formulate perceptual properties that are desirable in panoramic videos. The first two we call *per frame requirements* and the last two we call *temporal requirements*:

- 1) Each frame must be a good panoramic image;
- 2) Moving objects must be well preserved;
- 3) Temporal coherence for the scene should be observed. For example, in case 1, the background should not be projected to different positions along time, since the VP and the FOV are stationary;
- 4) Temporal coherence for the moving objects should also be observed. For example, in case 1, objects should not change size and orientation if these properties do not change in the viewing sphere.

#### C. The temporal viewing sphere and problem statement

We model the viewing information of a scene through time in the following way: let  $[0, t_0]$  be a time interval and

$$\begin{aligned} \mathbf{R} : [-\pi, \pi] \times [-\frac{\pi}{2}, \frac{\pi}{2}] \times [0, t_0] &\rightarrow \mathbb{R}^4 \\ (\lambda, \phi) &\mapsto (\mathbf{r}(\lambda, \phi), t) \end{aligned}$$

where  $\mathbf{r}$  was defined in section I-B. The image of  $\mathbf{R}$  we call the *temporal viewing sphere*. This set, denoted by  $\mathbf{TS}^2$ , is represented by the *equirectangular video*. An example of equirectangular video is shown in figure 11.

We now state the panoramic video as that of finding a projection

$$\begin{aligned} \mathbf{U} : S \subseteq \mathbf{TS}^2 &\rightarrow \mathbb{R}^3 \\ (\lambda, \phi, t) &\mapsto (U(\lambda, \phi, t), V(\lambda, \phi, t), t) \end{aligned}$$

with desirable properties.



Figure 11. Left: First frame of the equirectangular video we use in this section. Right: Last frame.

#### D. Solution for case 1

In this section we propose a solution for case 1: we show equations that model the temporal requirements and the discretization of them lead to energies that, when minimized among the image energies, lead to the final results. All the details can be found in [2], pp. 104-114, and the resulting videos are in [8].

A simple solution that satisfies desirable properties 1 and 2 would be generating a video where each frame is an optimizing image generated by the method we studied in section II. A result for this solution is shown in figure 12.



Figure 12. Left: Intermediate frame. Right: Last frame. The man becomes curved in the final frame due to the line constraints near him. Since he is not curved in the input temporal viewing sphere, this behavior shows temporal incoherence for this object. FOV: 180/120.

In [2], we present the following PDEs to model temporal coherence of the moving objects:

$$\begin{aligned} \frac{\partial U}{\partial \phi}(\lambda_{t_1, t_2}^{ob}(\lambda, \phi, t_1), \phi_{t_1, t_2}^{ob}(\lambda, \phi, t_1), t_2) - \frac{\partial U}{\partial \phi}(\lambda, \phi, t_1) &= 0 \\ \frac{\partial V}{\partial \phi}(\lambda_{t_1, t_2}^{ob}(\lambda, \phi, t_1), \phi_{t_1, t_2}^{ob}(\lambda, \phi, t_1), t_2) - \frac{\partial V}{\partial \phi}(\lambda, \phi, t_1) &= 0 \end{aligned}$$

Above  $(\lambda_{t_1, t_2}^{ob}, \phi_{t_1, t_2}^{ob})$  is a *transition function* that corresponds points of the objects between times  $t_1$  and  $t_2$ . Discretizing them and imposing them from one frame (time) to the next corrects the orientation problem seen in figure 12, but now all the scene starts to shake to satisfy the object constraints (see [8]). To correct this problem, we formulate equations that model the temporal coherence of the scene for this case:

$$\begin{aligned} \frac{\partial U}{\partial \phi}(\lambda, \phi, t_2) - \frac{\partial U}{\partial \phi}(\lambda, \phi, t_1) &= 0 \\ \frac{\partial V}{\partial \phi}(\lambda, \phi, t_2) - \frac{\partial V}{\partial \phi}(\lambda, \phi, t_1) &= 0 \end{aligned}$$

Using these equations to obtain an energy  $E_{sc}$  and minimizing it with the object energy  $E_{ob}$  (obtained from the object equations) and image energies lead to the final result in figure 13.

In [2], pp.115-116, we also present a preliminary solution for case 2. The final results are also in [8].

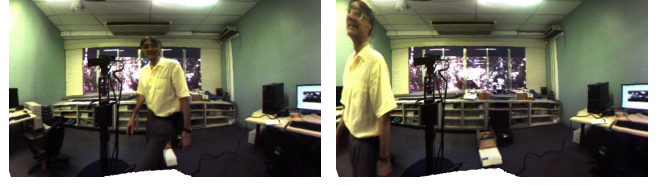


Figure 13. Left: Intermediate frame. Right: Last frame. Observe that the man is less curved and the scene is well preserved.

#### V. CONCLUSION AND FUTURE WORK

In this paper, we studied the panoramic image problem and proposed extensions related to it. The most important of them was the novel study about panoramic videos, which we intend to continue in the next years.

The next step in this study is to propose other solutions for cases 1 and 2 and compare them to the ones presented in this paper. Joining these two cases lead to solving the general case for stationary VP. Then we intend to move on to case 3, which seems more challenging.

Finally, we intend to apply all these ideas to real applications, such as cinema and sport broadcasting.

#### ACKNOWLEDGMENT

The authors would like to thank CNPq for the financial support.

#### REFERENCES

- [1] "Flickr: Equirectangular," Site address: <http://www.flickr.com/groups/equirectangular>.
- [2] L. K. Sacht, "Content-based projections for panoramic images and videos," Master's thesis, IMPA, 2010.
- [3] D. Zorin and A. H. Barr, "Correction of geometric perceptual distortions in pictures," in *SIGGRAPH '95: Proceedings of the 22nd annual conference on Computer graphics and interactive techniques*. New York, NY, USA: ACM, 1995, pp. 257-264.
- [4] M. Agrawala, D. Zorin, and T. Munzner, "Artistic multiprojection rendering," in *Proceedings of the Eurographics Workshop on Rendering Techniques 2000*. London, UK: Springer-Verlag, 2000, pp. 125-136.
- [5] L. Zelnik-Manor, G. Peters, and P. Perona, "Squaring the circles in panoramas," in *ICCV '05: Proceedings of the Tenth IEEE International Conference on Computer Vision*. Washington, DC, USA: IEEE Computer Society, 2005, pp. 1292-1299.
- [6] R. Carroll, M. Agrawal, and A. Agarwala, "Optimizing content-preserving projections for wide-angle images," *ACM Trans. Graph.*, vol. 28, no. 3, pp. 1-9, 2009.
- [7] P. Viola and M. J. Jones, "Robust real-time face detection," *Int. J. Comput. Vision*, vol. 57, no. 2, pp. 137-154, 2004.
- [8] L. K. Sacht, "Content-based projections for panoramic images and videos," url: [http://w3.impa.br/~leo-ks/msc\\_thesis](http://w3.impa.br/~leo-ks/msc_thesis).