

Automatic Upright Alignment of Multi-View Spherical Panoramas

Moritz Mühlhausen, Matthias Ueberheide, Marc Kassubeck, Leslie Wöhler, Marcus Magnor
<https://graphics.tu-bs.de/people/>

Institut für Computergraphik
 Technische Universität Braunschweig

Panorama images are now ubiquitously being used and are on their way to replace standard images. This makes it essential to adapt computer vision algorithms for panorama images. Without precise leveling of the cameras during acquisition, the acquired images have wavy horizons and slanted objects due to camera tilts which is usually not anticipated by algorithms for standard images. Hence, the first step for panorama alignment is to resolve these misorientations. This problem has been addressed before [2, 3] by introducing *Upright Adjustment* techniques, which level the horizon in photographs and panorama images. While this makes the adaptation of single image algorithms to panoramas easier, the next step is to extend algorithms for multiple images or image sequences. For this step another alignment problem arises: The cameras do not only need to be leveled properly, but the relative orientation needs to be known as well. In this work, we aim to align several panoramas to a reference panorama without proper alignment during acquisition. We focus only on the rotational component between panoramas, exploiting already known information from the *Upright Adjustment* step.

Given a set of K spherical panorama images with arbitrary rotations, we seek to rotate them in such a way that all images are leveled and face a common direction. For this their respective orthonormal coordinate systems $(\vec{x}, \vec{y}, \vec{z})$ need to be aligned, with \vec{y} facing upwards in the image. First, all images need to be rotated so that their north poles $\vec{y} = (0, 1, 0)$ are facing upwards and are perpendicular to the horizon within the image. These rotations do not depend on other images, so this can be computed for each image independently. To solve this, our work adopts the *Upright Adjustment* of Jung et al. [2], which updates the position of the north pole for a single image in an iterative process. A cost function is formulated to find the subsequent north pole position \vec{P} :

$$E(\vec{P}) = \alpha \sum_i (\vec{v}_i \cdot \vec{P})^2 + \beta \sum_j (\vec{h}_j \cdot \vec{P})^2 + \lambda (1 - \vec{y} \cdot \vec{P})^2$$

Here, \vec{v}_i are unit vectors perpendicular to great circles from vertical lines and \vec{h}_i are unit vectors from horizontal vanishing points, computed from horizontal line great circles. To prevent drastic change in a single iteration, the last term penalizes the deviation to the old north pole \vec{y} . α , β and λ are weighting parameters. The recovery of horizontal and vertical great circles, as well as the vanishing point detection, are utilizing a spherical hough transformation. Line segments are detected with an arbitrary line detection algorithm and categorized to be either horizontal or vertical. For each line segment the corresponding great circle is accumulated in spherical hough space to identify the best great circles. Afterwards, these are rasterized in spherical hough space to find vanishing points. With these information new north poles are computed subsequently, rotating the image after each iteration to obtain new great circles and vanishing points. The iterations are stopped when the difference between \vec{y} and \vec{P} becomes sufficiently small, taking \vec{P} as the final north pole.

After the *Upright Adjustment* step we have properly leveled panoramas. Since these still face different directions, they need to be rotated around the north poles \vec{y} . For the *Rotational Alignment* we use two upright adjusted images at a time, i.e. a panorama I and a reference panorama \hat{I} to align to. However, our method can handle any number of panoramas, by subsequently aligning all panoramas to the same reference \hat{I} . We keep the horizontal vanishing point hough space of the last iteration of the *Upright Adjustment*. Only the most dominant vanishing points based on their hough weights are used. These most dominant vanishing points are usually the most precise ones in the image, hence aligning the images accordingly increases accuracy. Moreover, our search area is restricted to the leveled horizon, reducing the dimension to 1D. For *Rotational Alignment* we need to find one pair of corresponding vanishing points and assign a new horizontal position to every pixel (n, m) of the image. However, points at infinity can be visually blocked by closer objects within one image. Hence, the surroundings of corresponding vanishing points can be very different, making matching difficult. Therefore, instead of matching the vanishing points directly, we compare the rotated images given a corresponding vanishing point pair. Assuming a vanishing point $\hat{h}_i \in \hat{H}$ of



Figure 1: After an *Upright Adjustment* step, our method automatically aligns a pair or more panoramas to face a common direction. Top: Original images, neither upright adjusted nor aligned. Bottom: Upright adjusted and aligned panorama images.

the reference image \hat{I} to correspond to $h_j \in H$ in the current image I , we rotate I by $\hat{h}_i - h_j$ around the north pole and obtain $I'_{\hat{h}_i, h_j}$.

$$I'_{\hat{h}_i, h_j}(n, m) = I(n, m - (\hat{h}_i - h_j))$$

Then, we compare the obtained image with \hat{I} to check whether or not the points actual correspond. We use the *Structural Similarity (SSIM)* for comparison instead of plain pixel values:

$$\max_{\hat{h}_i \in \hat{H}, h_j \in H} \left\| \text{SSIM}(\hat{I}, I'_{\hat{h}_i, h_j}) \right\|_2^2$$

Where \hat{H} is the set of the dominant vanishing points in the reference panorama \hat{I} and accordingly H is the set of dominant vanishing points in I . $\|\cdot\|_2^2$ is the squared norm, so we take the *SSIM* of every channel of the image separately and sum their squares. Testing with all vanishing point pairs \hat{h}_i, h_j , we obtain the best match. This is similar to the coarse alignment of Gurrieri et al. [1]. Yet, our advantage is that we only need to check a few rotations, instead of a complete search over the width of the image. Moreover our method is very robust, unless we found only different vanishing points, at least one of the rotations is correct.

For accuracy evaluation and stability tests synthetic data with 500 randomly chosen panorama horizontal shifts was used. We performed the *Rotational Alignment* and compared the known shift with the output of our method. The results contain no more than 4 pixels of misalignment per shift while more than half of all tested rotations are aligned within 1 pixel of the ground truth. Adding zero-mean gaussian white noise with a variance of 0.01 for further stability tests increased the maximum pixel misalignment to 10 pixels but for more than fifty percent of the tests the rotations are within 2 pixels to the perfect alignment. We also tested our approach on real-world scenes for *Rotational Video Stabilization* for panoramas with convincing results for alignment of key frames. Our method is very fast, as we utilize vanishing points obtained during *Upright Adjustment* only minimal additional time is required. Furthermore, our method is not restricted by the number of vanishing points and is therefore the first method to specifically target multi-view panorama setups, where the scenes do not follow the *Manhattan World Assumption*.

- [1] Luis E Gurrieri and Eric Dubois. Optimum alignment of panoramic images for stereoscopic navigation in image-based telepresence systems. In *Computer Vision Workshops (ICCV Workshops), 2011 IEEE International Conference on*, pages 351–358. IEEE, 2011.
- [2] Jinwoong Jung, Joon-Young Lee, Byungmoon Kim, and Seungyong Lee. Upright adjustment of 360 spherical panoramas. In *Virtual Reality (VR), 2017 IEEE*, pages 251–252. IEEE, 2017.
- [3] Hyunjoon Lee, Eli Shechtman, Jue Wang, and Seungyong Lee. Automatic upright adjustment of photographs with robust camera calibration. *IEEE transactions on pattern analysis and machine intelligence*, 36(5):833–844, 2014.