Automatic Upright Adjustment of Photographs

Hyunjoon Lee
POSTECH
Pohang, Korea
crowlove@postech.ac.kr

Eli Shechtman
Adobe Systems
Seattle, USA
elishe@adobe.com

Jue Wang
Adobe Systems
Seattle, USA
juewang@adobe.com

Seungyong Lee
POSTECH
Pohang, Korea
leesy@postech.ac.kr

Abstract

Man-made structures often appear to be distorted in photos captured by casual photographers, as the scene layout often conflicts with how it is expected by human perception. In this paper we propose an automatic approach for straightening up slanted man-made structures in an input image to improve its perceptual quality. We call this type of correction upright adjustment. *We propose a set of criteria for upright adjustment based on human perception studies, and develop an optimization framework which yields an optimal homography for adjustment. We also develop a new optimization-based camera calibration method that performs favorably to previous methods and allows the proposed system to work reliably for a wide variety of images. The effectiveness of our system is demonstrated by both quantitative comparisons and qualitative user studies.*

1. Introduction

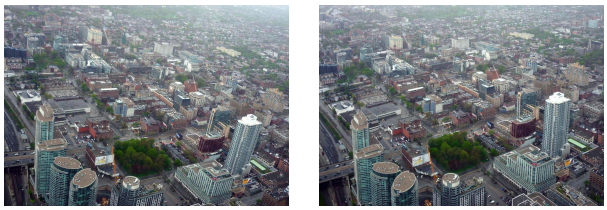
A large portion of consumer photos contain man-made structures, such as urban scenes with buildings and streets, and indoor scenes with walls and furniture. However, photographing these structures properly is not an easy task. Photos taken by amateur photographers often contain slanted buildings, walls, and horizon lines due to improper camera rotations, as shown in Fig. 1. On the contrary, our visual system always expects tall man-made structures to be straight-up, and horizon lines to be parallel to our eye level. This conflict leads us to a feeling of discomfort when we look at a photo containing slanted structures.

Assuming the depth variations of the scene relative to its distance from the camera are small, correcting a slanted structure involves a 3D rotation of the image plane. We call this type of correction *upright adjustment*, since its goal is to make man-made structures straight up as expected by human perception. Similar corrections have been known as *keystoning* and *perspective correction*, which can be achieved by manually warping the image using existing software such as Photoshop, or during capture using a spe-

(a) Urban building

(b) Planar board

(c) Indoor restaurant



(d) Urban scene

(e) Natural scene with mountains and trees

Figure 1. Various examples of upright adjustment of photos. (left) original; (right) our result.

cial Tilt-Shift lens[1]. However, the target domain of these tools is mostly facades of buildings, while our upright adjustment method does not explicitly assume specific types of objects in the scene. In addition, manual correction not only requires special skills, but also becomes tedious when we need to process hundreds of photos from a trip.

In this paper, we propose a fully automatic system for upright adjustment of photos. To the best of our knowledge, our system is the first one that automatically handles this kind of correction, although there have been several papers dealing with sub-problems of our framework. Our system introduces several novel technical contributions: (1) we propose various criteria to quantitatively measure the perceived quality of man-made structures, based on previous studies on human perception; (2) following the criteria, we propose an energy minimization framework to compute an optimal homography that can effectively minimize the perceived distortion of slanted structures; and (3) we propose a new camera calibration method which simultaneously estimates vanishing lines and points as well as camera parameters, and is more accurate and robust than the state-of-the-art. Although not designed to, our system is robust enough to handle some natural scenes as well (see Fig. 1e and additional results in the supplementary material). We evaluate the system comprehensively through both quantitative comparisons and qualitative user studies. Experimental results show that our system works reliably on a wide range of images without the need for user interaction.

1.1. Related work

Photo aesthetics and composition Automatic photo aesthetics evaluation tools [5, 11, 16, 7] and composition adjustment systems [15, 22] have been proposed recently, which introduced various criteria for aesthetics and composition quality of photographs. We propose a set of new criteria specific to the uprightness of man-made structures, based on well-known studies in human perception. Our method is based on an objective function that quantifies these criteria, and thus could potentially be used to enhance previous aesthetic evaluation methods with an additional uprightness score.

Manual correction Commercial software such as Adobe Photoshop provides manual adjustment tools, such as lens correction, 3D image plane rotation, and cropping. By combining these tools together, an experienced user can achieve similar upright adjustment results to our system. Professional photographers sometimes use an expensive Tilt-Shift lens for adjusting the orientation of the plane of focus and the position of the subject in the image for correcting slanted structures and converging lines. Both solutions require sophisticated interaction which is hard for a novice. Carroll *et al*. [2] proposed a manual perspective adjustment tool based on geometric warping of a mesh, which is more general than a single homography used in this paper, but requires accurate manual control.

Image rectification and rotation Robust methods for image rectification [14, 18, 23] were developed by analyzing the distortions of planar objects in the scene, such as windows of buildings. However, the main purpose of the methods was to use rectified objects in the image as input for other applications, such as 3D geometry reconstruction or texture mapping. Gallagher [9] proposed an automatic method that adjusts rotation of an image, but the method is limited to only in-plane rotations. Our method does not simply rectify or rotate the input but reproject overall scene of the image to obtain a perceptually pleasing result.

Camera calibration Calibrating camera parameters[2] from a single image is a well-studied problem [12, 10, 6, 20, 23, 17]. Most approaches employ a two-step framework: a set of vanishing points/lines is first extracted from the input image, and then used to calibrate the camera. In contrast, our system simultaneously estimates vanishing lines/points and camera parameters in a single optimization framework. Our technique is thus more robust and accurate than previous methods.

2. Adjustment Criteria and Formulation

In this section, we first discuss a set of criteria for upright adjustment of photos. We then propose our formulation of the image transformation used for upright adjustment.

2.1. Criteria

Scenes with well-structured man-made objects often include many straight lines that are supposed to be horizontal or vertical in the world coordinates. Our proposed criteria reflect these characteristics.

Picture frame alignment When looking at a big planar facade or a close planar object such as a painting, we usually perceive it as orthogonal to our view direction, and the horizontal and vertical object lines are assumed to be parallel and perpendicular to the horizon, respectively. When we see a photo of the same scene, the *artificial* picture frame imposes significant alignment constraints on the object lines, and we feel discomfort if the object line directions are not well aligned with the picture frame orientation [13, 8]. Figs. 1a and 1b show typical examples. It is also important to note

[1] http://en.wikipedia.org/wiki/Tilt-shift_photography

[2] Note that by *camera parameter calibration* we mean estimation of the intrinsic parameter and external orientation matrices of a camera, not radial and other non-linear distortions that are assumed in our work to be small or already corrected.

that such an artifact becomes less noticeable as the misalignments of line directions become larger, since in that case we begin to perceive 3D depths from a slanted plane.

Eye level alignment The eye level of a photo is the 2D line that contains the vanishing points of 3D lines parallel to the ground [4]. In a scene of an open field or sea, the eye level is the same as the horizon. However, even when the horizon is not visible, the eye level can still be defined as the connecting line of specific vanishing points. It is a well-known principle in photo composition that the eye level or horizon should be horizontal [8]. The eye level alignment plays an important role in upright adjustment especially when there exist no major object lines to be aligned to the picture frame. In Fig. 1d, the invisible eye level is dominantly used to correct an unwanted rotation of the camera.

Perspective distortion Since we do not usually see objects outside our natural field of view (FOV), we feel an object is distorted when the object is pictured as if it is out of our FOV [13, 4]. We can hardly see this distortion in ordinary photos, except those taken with wide-angle lenses. However, such distortion may happen if we apply a large rotation to the image plane, which corresponds to a big change of the camera orientation. To prevent this from happening, we explicitly constrain perspective distortion in our upright adjustment process.

Image distortion When we apply a transformation to a photo, image distortion cannot be avoided. However, human visual system is known to be tolerant to distortions of rectangular objects, while it is sensitive to distortions of circles, faces, and other familiar objects [13]. We consider this phenomenon in our upright adjustment to reduce the perceived distortions in the result image as much as possible.

2.2. Formulation

We assume no depth information is available for the input photo, and thus use a homography to transform it for upright adjustment. A more complex transformation could be adopted, e.g., content-preserving warping [2]. However such a transformation contains more degrees of freedom, and therefore requires a large amount of reliable constraints which should be fulfilled with user interaction or additional information about the scene geometry. A homography provides a reasonable amount of control to achieve visually plausible results in most cases, especially for man-made structures.

A given image can be rectified with a homography matrix $\mathbf{H}$ using the following equation [10]:

$$\mathbf{p}' = \mathbf{H}\mathbf{p} = \mathbf{K}(\mathbf{K}\mathbf{R})^{-1}\mathbf{p}, \tag{1}$$

where $\mathbf{p}$ and $\mathbf{p}'$ represent a position and its reprojection in the image, respectively. $\mathbf{K}$ and $\mathbf{R}$ are intrinsic parameter and orientation matrices of the camera, respectively:

$$\mathbf{K} = \begin{pmatrix} f & 0 & u_0 \\ 0 & f & v_0 \\ 0 & 0 & 1 \end{pmatrix} \text{ and } \mathbf{R} = \mathbf{R}_\psi \mathbf{R}_\theta \mathbf{R}_\phi,$$

where $\mathbf{R}_\psi$, $\mathbf{R}_\theta$, and $\mathbf{R}_\phi$ are rotation matrices with angles ψ, θ, and ϕ along the x, y, and z axes, respectively.

Although image rectification is useful for other applications, it can often generate a visually unpleasing result (see Fig. 6b). For upright adjustment, we modify Eq. (1) to obtain more flexible control for enhancing the perceptual quality of the results than a simple rectification. Our homography is defined by the following reprojection model:

$$\mathbf{p}' = \mathbf{H}\mathbf{p} = \mathbf{K}_1 \left\{ \mathbf{R}_1 (\mathbf{K}\mathbf{R})^{-1} \mathbf{p} + \mathbf{t}_1 \right\}, \tag{2}$$

where

$$\mathbf{K}_1 = \begin{pmatrix} f_{1x} & 0 & u_1 \\ 0 & f_{1y} & v_1 \\ 0 & 0 & 1 \end{pmatrix}, \tag{3}$$

$\mathbf{R}_1 = \mathbf{R}_{\psi_1} \mathbf{R}_{\theta_1} \mathbf{R}_{\phi_1}$, and $\mathbf{t}_1 = [t_{1x}\ t_{1y}\ 0]^T$.

Compared to Eq. (1), Eq. (2) contains a new intrinsic parameter matrix $\mathbf{K}_1$ with additional 3D rotation $\mathbf{R}_1$ and translation $\mathbf{t}_1$. This reprojection model implies re-shooting of the rectified scene using another camera placed at a possibly different position with novel orientation. We also allow this new camera to have different focal lengths in horizontal and vertical directions.

3. Adjustment Optimization

In this section, we derive and minimize an energy function for the image transformation formulated in Sec. 2.2 using the criteria defined in Sec. 2.1. We assume camera parameters $\mathbf{K}$ and $\mathbf{R}$ have been estimated by camera calibration. Then, we have 9 unknowns $f_{1x}, f_{1y}, u_1, v_1, \psi_1, \theta_1, \phi_1, t_x$ and t_y in Eq. (2). However, u_1 and v_1 simply shift the result image after the transformation, and we set $u_1 = u_0$ and $v_1 = v_0$. Our objective thus becomes optimizing Eq. (2) with respect to 7 parameters of $\mathbf{H}$.

Although other methods [12, 10] can also be used, we develop our own method for robust camera calibration, which will be presented in Sec. 4. In camera calibration, we take the *Manhattan world assumption*, *i.e.* the major line structures of the scene are aligned to the x-, y-, and z-directions in 3D. For example, a rectangular building is assumed to be oriented following the principal directions of the world.

As a result of camera calibration, in addition to $\mathbf{K}$ and $\mathbf{R}$, we obtain *Manhattan directions*, $\mathbf{M} = [\mathbf{v}_x\ \mathbf{v}_y\ \mathbf{v}_z]$, where $\mathbf{v}_x$, $\mathbf{v}_y$, and $\mathbf{v}_z$ represent the three vanishing points corresponding to the x-, y-, and z-directions, respectively. We

also obtain three pencils of vanishing lines, $\mathbf{L}_x$, $\mathbf{L}_y$, and $\mathbf{L}_z$, which contain 2D lines intersecting at vanishing points $\mathbf{v}_x$, $\mathbf{v}_y$, and $\mathbf{v}_z$, respectively. The vanishing lines in $\mathbf{L}_x$, $\mathbf{L}_y$ and $\mathbf{L}_z$ are projections of 3D lines that are parallel to the x-, y-, and z-axes, respectively.

3.1. Energy terms

Picture frame alignment For major line structures of the scene to be aligned with the picture frame, all vanishing lines corresponding to x- and y-directions should be horizontal and vertical in a photo, respectively. That is, vanishing lines in $\mathbf{L}_x$ and $\mathbf{L}_y$ should be transformed to horizontal and vertical lines by a homography $\mathbf{H}$, making vanishing points $\mathbf{v}_x$ and $\mathbf{v}_y$ placed at infinity in x- and y-directions, respectively.

Let $\mathbf{l}$ be a vanishing line, and $\mathbf{p}$ and $\mathbf{q}$ be two end points of $\mathbf{l}$. Then, the direction of the transformed line $\mathbf{l}'$ is:

$$\mathbf{d} = \frac{\mathbf{q}' - \mathbf{p}'}{\|\mathbf{q}' - \mathbf{p}'\|},$$

where

$$\mathbf{p}' = \frac{\mathbf{H}\mathbf{p}}{\mathbf{e}_z^T \mathbf{H}\mathbf{p}} \text{ and } \mathbf{q}' = \frac{\mathbf{H}\mathbf{q}}{\mathbf{e}_z^T \mathbf{H}\mathbf{q}}.$$

$\mathbf{e}_z = [0\ 0\ 1]^T$ is used to normalize homogeneous coordinates. We define the energy term as:

$$E_{pic} = \lambda_v \sum_i w_i \left(\mathbf{e}_x^T \mathbf{d}_{y_i}\right)^2 + \lambda_h \sum_j w_j \left(\mathbf{e}_y^T \mathbf{d}_{x_j}\right)^2, \quad (4)$$

where $\mathbf{d}_{y_i}$ is the direction of the transformed line $\mathbf{l}'_{y_i}$ of a vanishing line $\mathbf{l}_{y_i}$ in $\mathbf{L}_y$. $\mathbf{e}_x = [1\ 0\ 0]^T$, and $\mathbf{e}_x^T \mathbf{d}_{y_j}$ is the deviation of $\mathbf{l}'_{y_i}$ from the vertical direction. $\mathbf{d}_{x_i}$ is defined similarly for a vanishing line $\mathbf{l}_{x_j}$ in $\mathbf{L}_x$, and $\mathbf{e}_y = [0\ 1\ 0]^T$ is used to measure the horizontal deviation.

In Eq. (4), the weight w for a line $\mathbf{l}$ is the original line length before transformation, normalized by the calibrated focal length f, i.e., $w = \|\mathbf{q} - \mathbf{p}\|/f$. The weights λ_v and λ_h are adaptively determined using initial rotation angles, as the constraint of picture frame alignment becomes weaker as rotation angles get bigger. We use:

$$\lambda_v = \exp\left(-\frac{\psi^2}{2\sigma_v^2}\right) \text{ and } \lambda_h = \exp\left(-\frac{\theta^2}{2\sigma_h^2}\right), \quad (5)$$

where ψ and θ are calibrated rotation angles along x- and y-axes respectively. σ_v and σ_h are parameters to control the tolerances to the rotation angles. We fix them as $\sigma_v = \pi/12$ and $\sigma_h = \pi/15$ in our implementation.

Eye-level alignment The eye-level in a photo is determined as a line connecting two vanishing points $\mathbf{v}_x$ and $\mathbf{v}_z$ [4]. Let $\mathbf{v}'_x$ and $\mathbf{v}'_z$ be the transformed vanishing points:

$$\mathbf{v}'_x = \frac{\mathbf{H}\mathbf{v}_x}{\mathbf{e}_z^T \mathbf{H}\mathbf{v}_x} \text{ and } \mathbf{v}'_z = \frac{\mathbf{H}\mathbf{v}_z}{\mathbf{e}_z^T \mathbf{H}\mathbf{v}_z}.$$

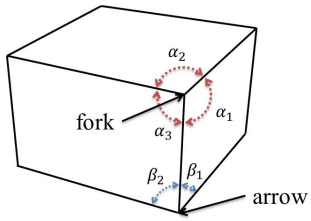


Figure 2. Perkins's law. Vertices of a cube can be divided into two categories; fork and arrow junctures. For a fork juncture, α_1, α_2, and α_3 should be greater than $\pi/2$. For an arrow juncture, both β_1 and β_2 should be less than $\pi/2$, and sum of the two angles should be greater than $\pi/2$. Vertices that violate the above conditions will not be perceived as vertices of a cube to the viewer.

Our objective is to make the eye-level horizontal, and the energy term is defined as:

$$E_{eye} = \left(\sum_i w_i + \sum_j w_j\right) \left(\mathbf{e}_y^T \mathbf{d}_e\right)^2,$$

where $\mathbf{d}_e = (\mathbf{v}'_z - \mathbf{v}'_x) / \|\mathbf{v}'_z - \mathbf{v}'_x\|$, and w_i and w_j are weights used in Eq. (4). Since eye-level alignment should be always enforced even when a photo contains lots of vanishing lines, we weight E_{eye} by the sum of line weights to properly scale E_{eye} with respect to E_{pic}.

Perspective distortion Perspective distortion of a cuboid can be measured using Perkins's law [13], as illustrated in Fig. 2. To apply it, we have to detect corner points that are located on vertices of a cuboid. We first extract points where the start or end points of vanishing lines from two or three different axes meet. We then apply the mean-shift algorithm [3] to those points to remove duplicated or nearby points. We also remove corner points with too small corner angles. Fig. 3 shows a result of this method.

We use the extracted corner points to measure perspective distortion under Perkins's law. For each corner point, we draw three lines connecting it to the three vanishing points. We then measure angles between the three lines to see if Perkins's law is violated or not:

$$^\forall \mathbf{c}_i, \ \min\left(\alpha_{i_1}, \alpha_{i_2}, \alpha_{i_3}\right) > \frac{\pi}{2} \quad (6)$$

where $\mathbf{c}_i$ represents a corner point. We only considers fork junctures, since arrow junctures can be transformed to fork junctures by swapping the direction of an edge.

Image distortion To accurately measure image distortion, we should detect circles and other important features in the input photo, which is a hard problem. We instead use an approximation in our system.

We first detect low-level image edges using Canny detector [1], then remove edge pixels that are nearby straight

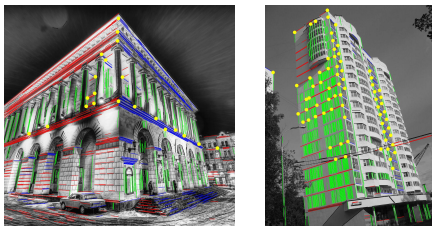


Figure 3. Results of our corner point extraction. Extracted points are marked as yellow dots.

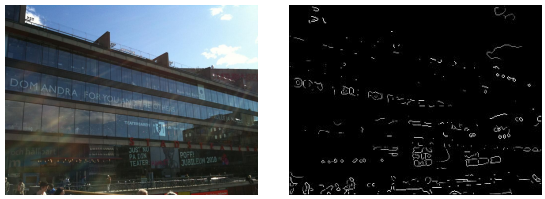


Figure 4. Feature detection: (left) original; (right) detected curved edge pixels. Some important features have been detected, such as human heads and letters, which should not be distorted.

lines. Assuming the remaining edge pixels are from curved lines that could be originated from some features (see Fig. 4), we measure distortions of these pixels using the following Jacobian measure:

$$E_{reg} = \lambda_r \sum_i \left\{ \det\left(J\left(\frac{\mathbf{H}\mathbf{p}_i}{\mathbf{e}_z^T \mathbf{H}\mathbf{p}_i} \right) \right) - 1 \right\}^2,$$

where $\mathbf{p}_i$ is a remaining edge pixel, $J(\cdot)$ is the Jacobian matrix, and $\det(\cdot)$ is the determinant. This energy increases when non-rigid transforms are applied to the pixels causing distortions of features. For λ_r, we used a small value 10^{-4}.

Focal length difference Our reprojection model for a homography allows different focal lengths along x- and y-axes for more natural results. However, we do not want the two lengths to differ too much. To enforce this property, we define the following energy:

$$E_{focal} = \lambda_f (f_{1x} - f_{1y})^2,$$

where we set $\lambda_f = (4/f)^2$ in our implementation.

3.2. Energy function minimization

Combining all the energy terms, the energy function we want to minimize for upright adjustment becomes:

$$\arg\min_{\mathbf{H}} \; E_{pic} + E_{eye} + E_{reg} + E_{focal} \qquad (7)$$
$$\text{subject to Eq. (6).}$$

To initialize the variables, we use $f_{1x} = f_{1y} = f$, $\psi_1 = 0$, $\theta_1 = 0$, $\phi_1 = -\phi$, and $t_x = t_y = 0$, where f and ϕ are values obtained by camera calibration.

Note that this energy function is non-linear and cannot be solved in a closed form. In practice, we use the numerical approach using `fmincon` in Matlab to minimize the energy function. Although global optimum is not guaranteed, this approach works quite well in practice.

4. Camera Calibration

In this section, we present a robust method for camera calibration to estimate the matrices $\mathbf{K}$ and $\mathbf{R}$ in Eq. (2). In previous methods [12, 10], Manhattan directions $\mathbf{M}$ are first determined using vanishing lines and vanishing points detected from the input photo, and then $\mathbf{K}$ and $\mathbf{R}$ are directly obtained from $\mathbf{M}$. However, in determining $\mathbf{M}$, corresponding vanishing points for x-, y-, and z-axes may not be obvious, because there could be many vanishing points possibly with position errors. The inaccuracy of $\mathbf{M}$ is then immediately propagated to $\mathbf{K}$ and $\mathbf{R}$. In contrast, our method estimates $\mathbf{K}$, $\mathbf{R}$, and $\mathbf{M}$ simultaneously using a MAP approach, and produces more reliable results.

Line segment detection Line segments are basic primitives in our calibration method. From the input image, we extract a set of line segments $\mathbf{L}$, using the method of von Gioi *et al*. [21] in a multi-scale fashion [20]. For each line segment $\mathbf{l}_i$, we store its two end points $\mathbf{p}_i$ and $\mathbf{q}_i$.

4.1. Calibration formulation

The joint probability of Manhattan directions $\mathbf{M}$, intrinsic matrix $\mathbf{K}$, and orientation matrix $\mathbf{R}$ with respect to line segments $\mathbf{L}$ can be expressed as follows:

$$\begin{aligned} p(\mathbf{K}, \mathbf{R}, \mathbf{M}|\mathbf{L}) &\propto p(\mathbf{L}|\mathbf{K}, \mathbf{R}, \mathbf{M}) p(\mathbf{K}, \mathbf{R}, \mathbf{M}) \\ &= p(\mathbf{L}|\mathbf{M}) p(\mathbf{M}|\mathbf{K}, \mathbf{R}) p(\mathbf{K}) p(\mathbf{R}), \end{aligned} \qquad (8)$$

with assumptions that $\mathbf{K}$ and $\mathbf{R}$ are independent of each other and also independent of $\mathbf{L}$. By taking log probability, we can rephrase Eq. (8) into an energy function as:

$$E_{K,R,M|L} = E_K + E_R + E_{M|K,R} + E_{L|M}. \qquad (9)$$

Prior E_K To define the prior for $\mathbf{K}$, we take a similar approach to [19, 20]. We assume that the center of projection $\mathbf{c}_p = (u_0, v_0)$ is the image center $\mathbf{c}_I = (c_x, c_y)$, and that the focal length f is the image width W. E_K is then defined as:

$$E_K = \lambda_f \left(\frac{\max(W, f)}{\min(W, f)} - 1 \right)^2 + \lambda_c \left\| \mathbf{c}_p - \mathbf{c}_I \right\|^2.$$

We set λ_f as 0.04 and λ_c as $(10/W)^2$.

Prior E_R For the prior of $\mathbf{R}$, we assume that the orientation of the camera is aligned with the principal axes of the world, which is a reasonable assumption in most cases. We have:

$$E_R = \lambda_\psi \psi^2 + \lambda_\theta \theta^2 + \lambda_\phi \phi^2.$$

The three rotation angles are not weighted equally. Particularly, we found that the prior for ϕ (z-axis rotation) should be stronger to enforce eye-level alignment. We thus use $[\lambda_\psi, \lambda_\theta, \lambda_\phi] = [4/\pi, 3/\pi, 6/\pi]^2$.

Posterior $E_{M|K,R}$ If $\mathbf{K}$ and $\mathbf{R}$ are known, $\mathbf{M}$ can be estimated as:

$$\mathbf{M} = [\mathbf{v}_x\ \mathbf{v}_y\ \mathbf{v}_z] = (\mathbf{KR})\mathbf{I}_3,$$

where $\mathbf{I}_3 = [\mathbf{e}_x\ \mathbf{e}_y\ \mathbf{e}_z]$ is the identity matrix. Using this property, we formulate our energy function as follows:

$$E_{M|K,R} = \lambda_M \sum_{i\in\{x,y,z\}} \left[\cos^{-1}\left\{\mathbf{e}_i^T \frac{(\mathbf{KR})^{-1}\mathbf{v}_i}{\|(\mathbf{KR})^{-1}\mathbf{v}_i\|}\right\}\right]^2.$$

This energy function covers the orthogonality of Manhattan directions [19] and the prior for zenith [20]. λ_M is set as $(24/\pi)^2$ in our experiments.

Posterior $E_{L|M}$ This term measures the conformity of detected line segments to the estimated vanishing points. We prefer vanishing points for which more line segments could be parts of vanishing lines. Our energy function is

$$E_{L|M} = \lambda_L \sum_i \min\left\{d(\mathbf{v}_x, \mathbf{l}_i), d(\mathbf{v}_y, \mathbf{l}_i), d(\mathbf{v}_y, \mathbf{l}_i)\right\},$$

where $d(\cdot)$ is the distance between a vanishing point and a line. We use the distance definition in [19]:

$$d(\mathbf{v}, \mathbf{l}) = \min\left(\frac{|\mathbf{r}^T\mathbf{p}|}{\sqrt{r_1^2 + r_2^2}},\ \delta\right), \tag{10}$$

where $\mathbf{p}$ and $\mathbf{q}$ are two end points of $\mathbf{l}$ and

$$\mathbf{r} = \left(\frac{\mathbf{p}+\mathbf{q}}{2}\right) \times \mathbf{v} = [r_1\ r_2\ r_3]^T.$$

δ is the given maximum error value for which we used 1.75 in our implementation. We set λ_L to 0.02.

Dealing with missing vanishing points When we estimate $\mathbf{M}$, we cannot always find all three vanishing points. For robustness, our energy terms, $E_{M|K,R}$ and $E_{L|M}$, should be able to handle this case. For $E_{M|K,R}$, we set the energy to be zero for a missing vanishing point, assuming that the point is located at the position estimated using $\mathbf{K}$ and $\mathbf{R}$. For $E_{L|M}$, we let $d(\mathbf{v}_{miss}, \mathbf{l}_i)$ always be δ for all $\mathbf{l}_i$.

4.2. Iterative optimization of K, R, and M

With the energy terms defined above, directly optimizing Eq. (9) is difficult since it is highly non-linear. We therefore use an iterative approach to find an approximate solution.

In the iteration, we alternatingly optimize $\mathbf{K}$ and $\mathbf{R}$, and $\mathbf{M}$. If we fix $\mathbf{M}$, we can optimize Eq. (9) with $\mathbf{K}$ and $\mathbf{R}$ by:

$$\arg\min_{\mathbf{K},\mathbf{R}}\ E_K + E_R + E_{M|K,R}. \tag{11}$$

Similarly, optimization of $\mathbf{M}$ can be achieved by solving:

$$\arg\min_{\mathbf{M}}\ E_{M|K,R} + E_{L|M}. \tag{12}$$

For optimizing $\mathbf{K}$ and $\mathbf{R}$ given $\mathbf{M}$, our implementation uses `fminsearch` in Matlab. On the other hand, optimization of $\mathbf{M}$ is still hard even if we fix $\mathbf{K}$ and $\mathbf{R}$, since $E_{L|M}$ truncates distances to δ as defined in Eq. (10). To solve Eq. (12), we use a discrete approximation, inspired by [20].

From the line segments $\mathbf{L}$, we hypothesize a large set of vanishing points $\mathbf{V} = [\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n]$, where each element is computed as the intersection point of two randomly selected lines. Optimizing $\mathbf{M}$ thus becomes selecting vanishing points from $\mathbf{V}$ to minimize the energy in Eq. (12). For each element of $\mathbf{M} = [\mathbf{v}_x\ \mathbf{v}_y\ \mathbf{v}_z]$, we find a vanishing point in $\mathbf{V}$ that minimizes the energy while retaining the other two elements.

The iterative optimization process requires good initial values of $\mathbf{K}$, $\mathbf{R}$, and $\mathbf{M}$ to work properly. We first select a small subset $\mathbf{V}_c = \{\mathbf{v}_{c_1}, \mathbf{v}_{c_2}, \ldots, \mathbf{v}_{c_k}\}$ from $\mathbf{V}$ that is the "closest to all lines" in the following way:

$$\arg\min_{\{\mathbf{v}_{c_1},\ldots,\mathbf{v}_{c_k}\}} \sum_i \min\left\{d(\mathbf{v}_{c_1}, \mathbf{l}_i), \ldots, d(\mathbf{v}_{c_k}, \mathbf{l}_i)\right\},$$

where we set $k = 9$ in our implementation. We then add a special vanishing point $\mathbf{v}_{miss}$, representing a missing vanishing point, into $\mathbf{V}_c$ because $\mathbf{V}_c$ may not contain all Manhattan directions of the scene. For each triplet of vanishing points in $\mathbf{V}_c$, we optimize $\mathbf{K}$, $\mathbf{R}$, and $\mathbf{M}$ using Eqs. (11) and (12), and then evaluate Eq. (9). Finally, $\mathbf{K}$, $\mathbf{R}$, and $\mathbf{M}$ with the minimum energy are used as our calibration results.

Although initial $\mathbf{V}_c$ may not contain all Manhattan directions, the missing directions can be detected from $\mathbf{V}$ while optimizing Eq. (12) in the iterative optimization process. Optimizing $\mathbf{K}$, $\mathbf{R}$, and $\mathbf{M}$ for all possible triplets in $\mathbf{V}_c$ might be computationally expensive. Thus we use some early termination strategies for speedup. Details can be found in the supplementary material.

Grouping vanishing lines After the calibration process, we determine the vanishing lines for each vanishing point in $\mathbf{M}$. Three sets of vanishing lines, $\mathbf{L}_x$, $\mathbf{L}_y$, and $\mathbf{L}_z$, are obtained from $\mathbf{L}$ by:

$$\mathbf{L}_i = \{\mathbf{l} \in \mathbf{L} \mid d(\mathbf{v}_i, \mathbf{l}) < \delta\},\ i \in \{x, y, z\},$$

where $d(\cdot)$ is the distance function defined in Eq. (10). Examples of camera calibration results with estimated vanishing lines can be found in the supplementary material.

5. Results

We implemented our algorithms using Matlab. For experiments, we used a PC with Intel Core i7 CPU (no multithreading) and 6GB RAM. It took about 7∼20 seconds

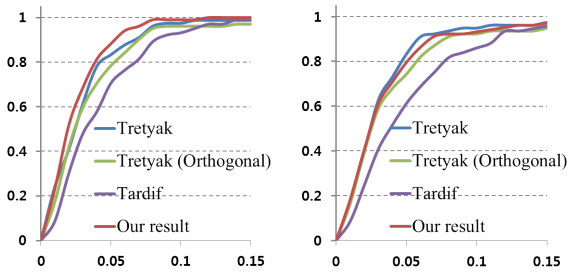


Figure 5. Cumulative histograms of the errors in eye-level estimation. (x-axis) eye-level estimation error; (y-axis) proportion of the images in the date set. See [20] for the details of the error metric.

to obtain the upright adjustment result for a tested image where the main chunks were: camera calibration (40% of the time), adjustment optimization and applying the homography (∼20% each) and line segment detection (∼10%). We downsized the input image to about 1M pixels for computing the homography $\mathbf{H}$, and applied the computed $\mathbf{H}$ to the original. All parameters were fixed in our experiments.

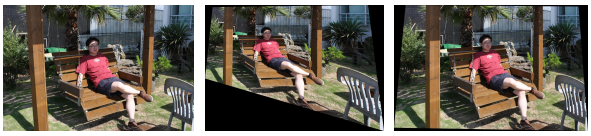
5.1. Evaluation of our camera calibration method

We compared our camera calibration method with several state-of-the-art techniques, Tardif [19], Tretyak *et al*. [20], and Mirzaei and Roumeliotis [17], using two datasets, York Urban [6] and Eurasian Cities [20]. For the results of Tardif [19] and Tretyak *et al*. [20], we used the implementations of the authors. For Mirzaei and Roumeliotis [17], we used the data given in the material provided by the authors.

Fig. 5 shows comparisons with Tardif and Tretyak *et al*. using the accuracy of the estimated eye-level angle, the measure adopted by the latter. We got better results than both methods on the York dataset. Our results were slightly worse than Tretyak *et al*. on the Eurasian Cities dataset, as their method was specifically targeted to eye-level estimation without reconstructing Manhattan directions. However, if the Manhattan world assumption is enforced in the method of Tretyak *et al*.[3], our results were better with the Eurasian Cities dataset as well. Furthermore, we obtained better results than Tardif using the measure of the focal length accuracy (see the supplementary material).

We also compared our method with Mirzaei and Roumeliotis [17], which assumes the ground truth $\mathbf{K}$ is known and that the scene has exactly three vanishing points (as in the York Urban dataset). Our method does not assume any of these. In comparison with the York Urban dataset, we could obtain comparable (without the ground truth $\mathbf{K}$) or better (with the ground truth $\mathbf{K}$) results. Details can be found in the supplementary material.

[3]The method of Tretyak *et al*. does not assume the Manhattan world and estimates the eye-level only, so we could not compare other quantity produced by our method, such as vanishing points and camera parameters.



(a) Original (b) Rectified (c) Our result

Figure 6. Adjustment of a photo with large camera rotations.

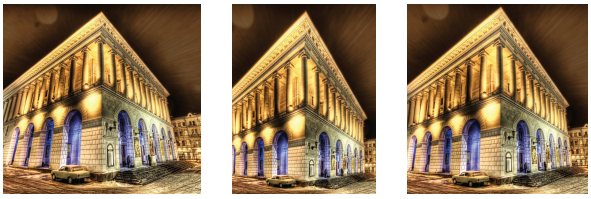


Figure 7. Perspective distortion control. (left) original; (middle) w/o perspective distortion constraint; (right) w/ the constraint.

5.2. Effects of upright adjustment criteria

Picture frame alignment is important for photos of big planar objects, such as facades of buildings and billboards. However, its effect should diminish as the rotation angles of the camera increase, otherwise it will lead to undesirable distortion (Fig. 6b). Note that if picture frame alignment dominates other criteria, the adjustment result becomes similar to simple image rectification. Our system automatically handles this problem with the adaptive weight scheme (Eq. (5)) as well as the perspective and image distortion criteria, generating a better result shown in Fig. 6c.

Eye-level alignment becomes more important as the effect of picture frame alignment gets weaker (Fig. 1d), although applying this criterion would always help obtain a better result. Perspective distortion control prevents too strong adjustment that could make objects in the image appear distorted (Fig. 7). We found that allowing the focal lengths in x- and y-directions to slightly deviate with Eq. (3), resulting in a small aspect ratio change, is often useful to ease the perspective distortion. We also found that artists do similar adjustments manually to make their results feel more natural in Keystone correction of photos [4].

5.3. Comparison and user study

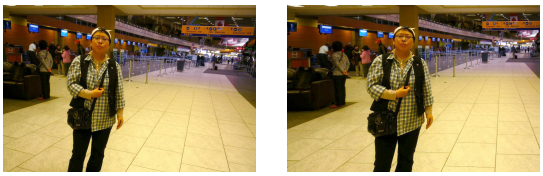
To compare with manual adjustment tools, we asked proficient Photoshop users to manually correct some examples with the explicit guidance of making the man-made structures upright. Our fully automatic results were similar to those of manual adjustments (see the supplementary). We also compared our results with photos taken using a Tilt-Shift lens. Our method could produce similar results for architectural scenes, but without using a special equipment that requires manual shift control (see Fig. 8).

To prove that our system does improve the perceptual quality of input images, we conducted a user study on Ama-

[4]http://youtu.be/QkG241258FE

Original Tilt-Shift Our result

Figure 8. Comparison with Tilt-Shift lens. A Canon TS-E 24mm f/3.5 L lens was used for taking the photo in the middle.



Original Result

Figure 9. A failure example. The human face and body have been distorted in the upright adjustment process.

zon Mechanic Turk. We used 40 pairs of original and adjusted images for the user study where 100 independent participants were asked to select the preferred image from each pair. On average our result was preferred in 72.4% of the cases. More details are in the supplementary material.

Limitations Our system uses a single homography to correct a photo under the uniform depth assumption for a scene. Although this assumption typically does not hold, in practice our method generates satisfying results for a wide range of images, due to the robustness of perspective perception [13]. However, for human faces or other important features in a photo, the adjusted result may contain noticeable distortion (see Fig. 9). In addition to that, our method might not produce an optimum result if the vanishing lines are incorrectly detected or grouped in a wrong way.

6. Conclusion

We proposed an automatic system that can adjust the perspective of an input photo to improve its visual quality. To achieve this, we first defined a set of criteria based on perception theories, then proposed an optimization framework for measuring and adjusting the perspective. Experimental results demonstrate the effectiveness of our system as an automatic tool for upright adjustment of photos containing man-made structures.

As future work, we plan to incorporate additional constraints to avoid perspective distortions on faces or circles. We also plan to extend our method to video.

Acknowledgments

This work was supported in part by Industrial Strategic Technology Development Program of KEIT (KI001820), IT/SW Creative Research Program of NIPA (NIPA-2011-C1810-1102-0030), and Basic Science Research Program of NRF (2010-0019523).

References

[1] J. Canny. A computational approach to edge detection. *IEEE PAMI*, 8:679–698, 1986.

[2] R. Carroll, A. Agarwala, and M. Agrawala. Image warps for artistic perspective manipulation. *ACM TOG*, 29:1, 2010.

[3] D. Comaniciu and P. Meer. Mean Shift: A robust approach toward feature space analysis. *IEEE PAMI*, 24:603–619, 2002.

[4] J. D'Amelio. *Perspective Drawing Handbook*. Dover Publications, 2004.

[5] R. Datta, D. Joshi, J. Li, and J. Z. Wang. Studying aesthetics in photographic images using a computational approach. In *Proc. ECCV*, 2006.

[6] P. Denis, J. H. Elder, and F. J. Estrada. Efficient edge-based methods for estimating Manhattan frames in urban imagery. In *Proc. ECCV*, 2008.

[7] S. Dhar, V. Ordonez, and T. L. Berg. High level describable attributes for predicting aesthetics and interestingness. In *Proc. CVPR*, 2011.

[8] M. Freeman. *The Photographer's Eye: Composition and Design for Better Digital Photos*. Focal Press, 2007.

[9] A. Gallagher. Using vanishing points to correct camera rotation in images. In *Proc. Computer and Robot Vision*, 2005.

[10] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*, chapter 8, pages 213–229. Cambridge University Press, 2004.

[11] Y. Ke, X. Tang, and F. Jing. The design of high-level features for photo quality assessment. In *Proc. CVPR*, 2006.

[12] J. Kosecka and W. Zhang. Video compass. In *Proc. ECCV*, 2002.

[13] M. Kubovy. *The Psychology of Perspective and Renaissance Art*. Cambridge University Press, 2003.

[14] D. Liebowitz and A. Zisserman. Metric rectification for perspective images of planes. In *Proc. CVPR*, 1998.

[15] L. Liu, R. Chen, L. Wolf, and D. Cohen-Or. Optimizing photo composition. *Computer Graphic Forum*, 29:469–478, 2010.

[16] Y. Luo and X. Tang. Photo and video quality evaluation: Focusing on the subject. In *Proc. ECCV*, 2008.

[17] F. M. Mirzaei and S. I. Roumeliotis. Optimal estimation of vanishing points in a Manhattan world. In *Proc. ICCV*, 2011.

[18] P. Muller, G. Zeng, P. Wonka, and L. J. V. Gool. Image-based procedural modeling of facades. *ACM TOG*, 26:85, 2007.

[19] J.-P. Tardif. Non-iterative approach for fast and accurate vanishing point detection. In *Proc. ICCV*, 2009.

[20] E. Tretyak, O. Barinova, P. Kohli, and V. Lempitsky. Geometric image parsing in man-made environments. *IJCV*, pages 1–17, 2011.

[21] R. von Gioi, J. Jakubowicz, J.-M. Morel, and G. Randall. LSD: A fast line segment detector with a false detection control. *IEEE PAMI*, 32:722–732, 2010.

[22] L.-K. Wong and K.-L. Low. Saliency retargeting: An approach to enhance image aesthetics. In *Proc. WACV*, 2011.

[23] Z. Zhang, Y. Matsushita, and Y. Ma. Camera calibration with lens distortion from low-rank textures. In *Proc. CVPR*, 2011.