Robust integral image rectification framework using perspective transformation supported by statistical line segment clustering

E. T. Koufogiannis,* N. P. Sgouros and M. S. Sangriotis

Department of Informatics and Telecommunications, University of Athens Panepistimiopolis, Athens, Greece 15784

*Corresponding author: efthimis@di.uoa.gr

Received 5 August 2011; revised 19 October 2011; accepted 2 November 2011;
posted 4 November 2011 (Doc. ID 152277); published 5 December 2011

In most integral image analysis and processing tasks, accurate knowledge of the internal image structure is required. In this paper we present a robust framework for the accurate rectification of perspectively distorted integral images based on multiple line segment detection. The use of multiple line segments increases the overall fault tolerance of our framework providing strong statistical support for the rectification process. The proposed framework is used for the automatic rectification, metric correction, and rotation of distorted integral images. The performance of our framework is assessed over a number of integral images with varying scene complexity and noise levels. © 2011 Optical Society of America

OCIS codes: 100.3020, 100.5010, 100.6890, 110.2960, 110.2990.

1. Introduction

The technological advances in sensor and display technologies as well as in optical component manufacturing allowed the development of three-dimensional (3D) capturing and display devices. As these devices are now penetrating consumer market applications, there is a need for improving them and increasing their robustness. In this paper we focus on integral imaging (InIm) systems. The InIm principle that was initially formulated by the Nobel Laureate G. Lippman [1] back in 1908 is currently considered one of the most promising techniques for delivering 3D content, featuring full color, adequate detail, and depth levels and support for multiple simultaneous viewers [2]

The basic InIm acquisition setup depicted in Fig. 1(a) comprises a charged coupled device (CCD) and a lens array (LA) [3,4] On the receiver, a typical InIm display setup is assembled of a liquid crystal display (LCD) and an appropriate LA, as depicted in Fig. 1(b).

Each of the lenses in the LA grid placed over the CCD of the acquisition device forms an elemental image (EI) on the sensor, which is a micrography of a part of the scene. The projection of the EIs through the LA on the receiver integrates the EIs in the space between the display and the viewer reproducing the initial 3D scene.

Because of misalignments that may occur when the LA is placed over the CCD of the acquisition device, perspective distortions may be introduced in the EI grid, which transform the square shaped EIs into convex quadrilaterals. Figure 2(a) shows a two-dimensional (2D) rendered image of a dice while Fig. 2(b) shows the ideally acquired InIm (I_{id}) where no LA-CCD misalignments are present. A misalignment of the LA and the CCD results in a geometric integrity degradation of the acquired InIm (I_{ac}), as shown in Fig. 2(c). These misalignments finally yield a poor 3D representation through an InIm display or a deformed 3D object after applying a reconstruction process like the one presented in [5]. Consequently an InIm rectification stage is required to correct these perspective distortions [6] on the acquired image.

This need was initially identified and accurately resolved in 2006 [7] accounting for small rotational

---

0003-6935/11/34H265-13$15.00/0

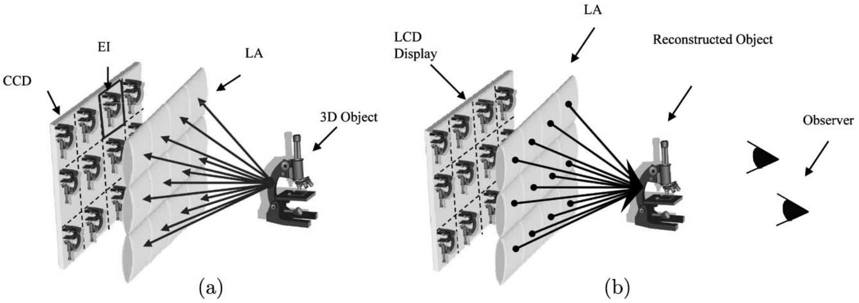


Fig. 1. (a) Integral imaging acquisition, (b) InIm display setup.

misalignments and imperfections of the LA grid on a 2D plane parallel to the CCD sensor. A more generic approach was recently proposed [8] to compensate for perspective distortions due to LA and CCD misalignments.

In this paper we propose an alternative approach to the work presented in [8], targeting an automated and fault tolerant detection and rectification framework with strong statistical support, based on the InIm characteristics. The basic concept of the proposed framework is to utilize the EIs' boundaries in the rectification process as a global InIm feature that localizes an orthogonal grid present in the InIm structure. This process increases the reliability and robustness of the algorithm, taking into account the whole InIm structural components.

In detail, the proposed framework uses a robust multiple line segment detection algorithm (LSD) [9] to detect and register line segments in the InIm belonging to the EI boundaries. Consecutively the registered segments are clustered according to their collinearity. The collinearity clustering approach is used to identify and remove line segments not belonging to the boundaries of the EIs. The groups of the remaining line segments that are highly likely to belong to the boundaries of EIs are used in a least square fitting sense to reconstruct the grid lines in the distorted InIm. The reconstructed grid lines are grouped into two concurrent line sets that converge into two points in the InIm plane. These two estimated points are called vanishing points [6] and are used in the rectification process, which follows immediately after the determination of the best pair of vanishing points.

The described approach outperforms methods based on a combination of an edge detector and the Hough transform in terms of registration accuracy and speed. It should be noted that the accuracy of the InIm rectification process is further increased due to the fact that the statistical derivation of the parameters is based on the whole EI grid registration. The described process does not require any manual parameter setting; therefore automated rectification can be preformed without any user interaction.

The rest of this paper is organized in four sections. In Section 2 we present the details of LSD and the clustering method used for ruling out invalid line segments. In the same section the concurrent line sets are produced. In Section 3 we analyze the basic principles of the rectification process used in our work and propose certain augmentations to further increase the performance of the algorithm in real applications. In Section 4 we provide the results of the application of the framework in a set of different InIms to evaluate the accuracy and noise robustness of the algorithm. Section 5 concludes this paper, summarizing the outcomes of the present research.

2. EIs Grid Line Reconstruction

In this section we analyze the procedure applied in the acquired InIm, I_{ac}, shown in Fig. 2(c), for the detection and the registration of the grid of straight lines formed by the EI boundaries. This process is a fundamental step required for the accurate InIm rectification described in Section 3. The above process is divided into three distinct steps that are further elaborated in this section.

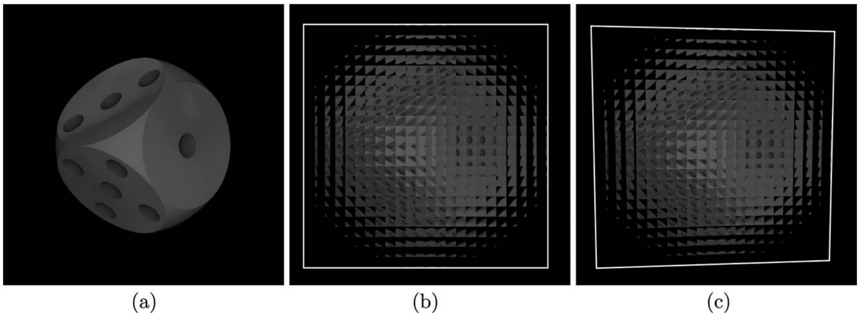

(a) (b) (c)

Fig. 2. (a) 2D rendered image of a dice. (b) An undistorted InIm of the dice, I_{id}. (c) A distorted InIm of the dice, I_{ac}. The InIm borders are shown for illustration purposes.

A. Detection and registration of the InIm's line segments, aiming to reconstruct the EI's boundaries.

B. Rejection of the majority of line segments that do not belong to EI's boundaries.

C. Clustering of the remaining segments into collinear groups and estimation of the line equation for each group.

A. Detection and Registration of InIm Line Segments

The first step toward the registration of the straight lines of the grid is the detection and registration of the line segments that belong to the EI boundaries in I_{ac}. For this purpose we use the LSD algorithm, which has a number of strong features, as it does not require any parameter setting and is not deceived by the random texture [9] of the contents of the EIs. In addition the LSD algorithm has linear execution times, making it an ideal candidate for a fast InIm rectification framework.

The LSD algorithm uses a region growing process followed by an estimation of the line support region for each line segment. Finally, an adaptation of the method presented in Desolneux [10] is used for line segment validation. In detail:

i. A region growing process uses as a feature the angle between the perpendicular direction to each pixel gradient and the level-line to form line support regions. The pixels assigned to each line support region share the same level-line angles up to a certain precision.

ii. Line segments are formed using the pixels in line support regions. The gradient magnitude of the pixels is used as the pixels mass and each line support region is approximated by a rectangle representing the line segment. The center of the rectangle is assigned to the center of mass of the pixels in the region and the orientation of the rectangle is determined by the first inertia axis of the pixels in the region. The size of the rectangle is adjusted to contain all the pixels in the region.

iii. Each of the resultant rectangles is refined using a number of false alarms (NFA) approach. The rectangular approximation for each line support region is the one with the smallest NFA value.

The algorithm outputs a sequence of registered line segments $\{l_i\}$ where each segment l_i is specified by its start and end points (s_i, e_i), denoted by their coordinates (x_{s_i}, y_{s_i}) and (x_{e_i}, y_{e_i}), respectively.

B. Rejection of Falsely Detected Boundary Segments

The detected and registered line segments after applying the LSD algorithm are shown in Fig. 3(a). As can be easily noted in Fig. 3(a), the sequence $\{l_i\}$ contains segments that belong to the grid lines, but it also contains segments that belong to the EI content. A careful examination of Fig. 3(a) shows that the direction angles of the segments belonging to the grid lines form two populations separated by an angular distance around 90°, while the rest of the line segments have random direction angles.

This feature is used in this step of the algorithm to reject most of the undesired line segments. In detail the slope of each segment in $\{l_i\}$ is initially calculated and its corresponding direction angle sequence is evaluated. Consequently the histogram of the direction angle values is calculated and plotted in the range $(-45°, 135°)$ as shown in Fig. 4(a) to avoid folding around 90°.

Each of the two lobes in the resulting histogram shown in Fig. 4(a) corresponds to segments parallel to the lines forming the EI grid. The rest of the histogram bins contain small populations of line segments that represent the content within the EIs. Accounting for small perspective distortions the population of the lobe closer to 0° is labeled h, corresponding to the horizontal grid lines of I_{id}, and the population closer to 90° is labeled ν, corresponding to the vertical grid lines of I_{id}.

In order to automatically discriminate between the two segment groups h and ν, as well as segments that belong to the EI content, we developed the following process:

i. We use an optimum threshold algorithm (OTA) [11] to partition the given histogram into two sets of consecutive bins in such a way that separability of the corresponding populations $\{l_{hi}\}$ and $\{l_{vi}\}$ is maximized.

ii. For each population we form a new histogram and calculate the bin having the maximum count. We repartition each population based on the calculated maximum value [Fig. 4(b), 4(c)].

iii. We successively apply the OTA on each of the repartitioned populations of step ii to calculate two

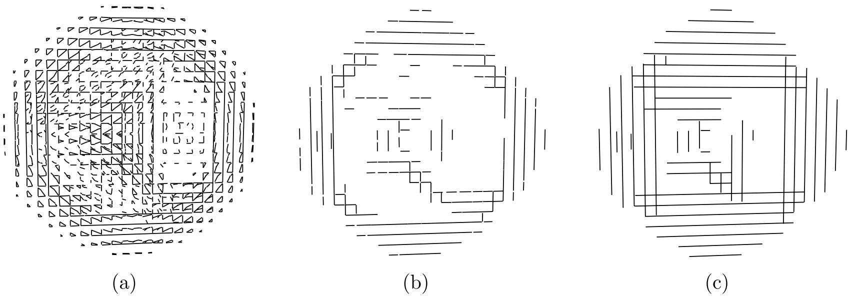
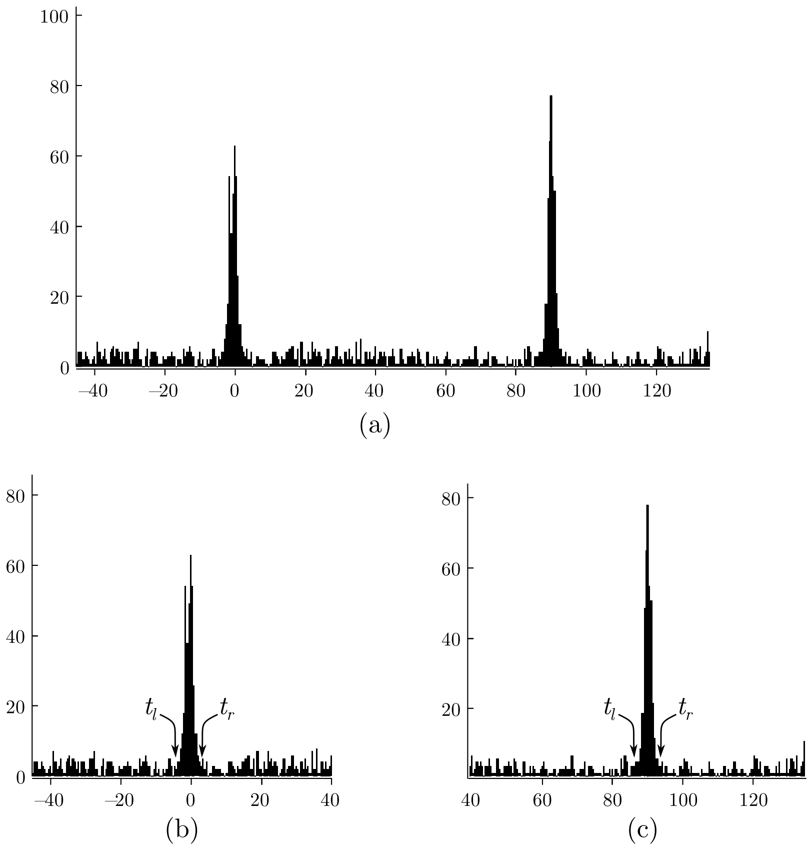


Fig. 3. (a) Line segments detected using the LSD. (b) Remaining line segments after rejecting most of the undesired line segments. (c) Line parts fitted on the clustered line segments.



(a)

(b)

(c)

Fig. 4. Histogram of the direction angles (a) of all detected line segments using the LSD algorithm, (b) of the line segments corresponding to horizontal lines, and (c) of the line segments corresponding to vertical lines.

thresholds t_l and t_r for each of the populations $\{l_{hi}\}$ and $\{l_{vi}\}$ that separate these populations from the rest of the initial histogram containing undesired line segments.

iv. Finally, we disregard all segments with lengths less than 0.1 of the maximum segment length to minimize the uncertainty introduced by small noisy segments.

The two line segment groups S_h and S_v resulting after applying the above process are plotted in Fig. 3(b). By direct comparison between Fig. 3(a) and Fig. 3(b) it can be easily noticed that most of the line segments that do not belong to the EI boundaries are ruled out. A small number of line segments that are misregistered or belong to the EI content still remain at this point. The error introduced by these line segments will be further reduced at the next stage of the proposed framework.

C. Clustering of Collinear Segments

In order to further reduce the remaining error from the previous stage, collinear segments are automatically clustered and each cluster is replaced by a straight line fitted to the data in a least squares sense.

In detail, a hierarchical clustering procedure is used in order to automatically classify the line segments of each group into collinear clusters. For this purpose we define a symmetric dissimilarity measure $d_m(l_i, l_j)$ [12] for every pair of segments (l_i, l_j) as shown in Fig. 5(a). Let (s_i, e_i) and (s_j, e_j) be the end points of these segments, respectively. Then, $d_m(l_i, l_j)$ is defined as

$$d_m(l_i, l_j) = \max\{d(s_i, l_j), d(e_i, l_j), d(s_j, l_i), d(e_j, l_i)\}, \tag{1}$$

where $d(p, l)$ represents the distance of the point p from the line segment l. Moreover the definition of $d_m(l_i, l_j)$ is extended to the dissimilarity measure between two clusters of collinear segments C_p and C_q, as

$$d_m(C_p, C_q) = \min\{d_m(l, l')\}, \qquad l \in C_p, \qquad l' \in C_q. \tag{2}$$

A variation of the hierarchical clustering technique, [13] called single-linkage clustering, is used to formulate the different clusters and the procedure terminates as soon as the minimum dissimilarity measure of the clusters exceeds a threshold value d_{th}. The value for d_{th} is automatically calculated through the following process:

i. The dissimilarity measure for every pair of registered line segments is computed, and the histogram for the resulting sequence $\{d_s(n)\}$ is constructed as shown in Fig. 5(b).

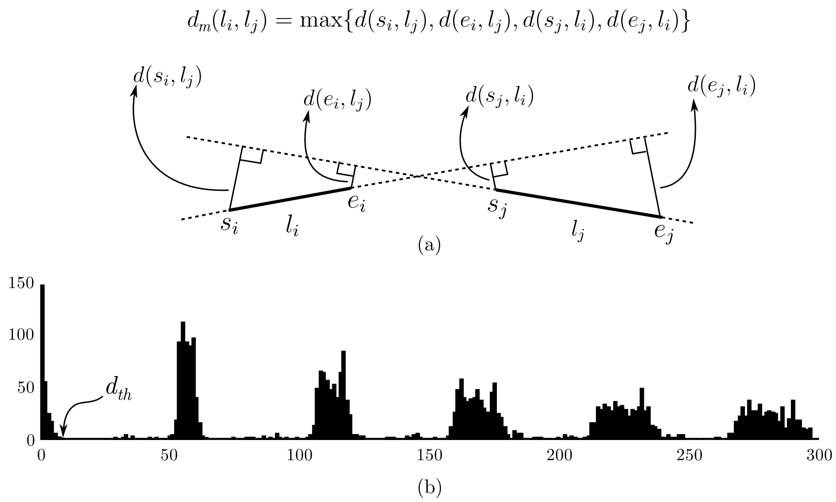


Fig. 5. (a) Geometric representation of the dissimilarity measure $d_m(l_i, l_j)$ between two segments. (b) The histogram of $\{d_s(n)\}$. Values in the first lobe correspond to collinear segments.

ii. The first empty histogram bin is found and d_{th} is set equal to the maximum value of the previous bin population.

This choice for d_{th} is justified through the fact that the dissimilarity measure between collinear and noncollinear segments presents significantly different values. Hence it is expected that the histogram of Fig. 5(b) will be characterized by gaps, formed by a number of empty bins between different groups of $\{d_s(n)\}$. After clustering, a straight line is fitted to the segments belonging to each collinear group having the minimum mean distance from the end points of these segments.

In Fig. 3(c), the fitted lines are plotted. The plotted lines are limited to the extent of the line segments in each cluster for illustration purposes. At this point two groups of line equations are derived, denoted as L_h and L_v, which will be used in the InIm rectification process.

3. InIm Rectification Process

In this section we introduce the mathematical model that was used to formalize the rectification process. Moreover we provide the implementation details for the InIm rectification case. The proposed algorithm reconstructs an estimate of the ideal InIm I_{id} using the acquired InIm I_{ac} and the two groups of estimated grid lines L_h and L_v provided by the process described in Section 2.

3.A. Mathematical Model

As already mentioned in the introductory part of this paper, the ideal I_{id} and the acquired InIm I_{ac} are related through a perspective transformation. Suppose that the grid lines of I_{ac} have been precisely detected, registered, and plotted as shown in Fig. 6. This figure shows how these lines are transformed through a sequence of transformations, resulting in a square grid with horizontal and vertical lines in I_{id}. It should be noted that in Fig. 6, I_{ac} is depicted with an exaggerated perspective distortion in order to make apparent some important properties of the distorted grid.

According to the theory [6] each group of parallel lines of the square grid in I_{id} has been transformed into a set of concurrent lines in I_{ac}, which intersect at a single point, called vanishing point. The two vanishing points V_1 and V_2 are also shown in Fig. 6. The line defined by the two vanishing points is called vanishing line. Since I_{id} is a similarity transformation of I_{si} and I_{si} is an affine transformation of I_{af}, then I_{si} has a square grid like I_{id}, whereas I_{af}'s grid consists of parallelograms. The perspective transform that projects a point in I_{ac} to a point in I_{id} is given by Eq. (3):

$$\mathbf{x} = H\tilde{\mathbf{x}}, \tag{3}$$

where H is a 3×3 real value matrix denoting the above mentioned perspective transformation, and $\mathbf{x}$ and $\tilde{\mathbf{x}}$ denote the homogeneous three component coordinate vectors for two related points in I_{id} and I_{ac}, respectively.

According to Liebovitz [14], the perspective transformation H can be considered as the product of three matrices of the same size given by Eq. (4)

$$H = H_s H_a H_p, \tag{4}$$

where H_p, H_a, and H_s represent a pure perspective, an affine, and a similarity transformation, respectively. To this end, I_{id} results after sequentially computing two intermediate images, I_{af}, and I_{si}, applying transformations of H_p, H_a, and H_s, as shown in Fig. 6. This transformation breakdown structure constitutes a simple approach for the estimation of the resultant transformation matrix H by separately evaluating each component matrix based on some prescribed geometric relations between the corresponding images.

According to [14] the pure perspective transformation matrix H_p is given by Eq. (5):

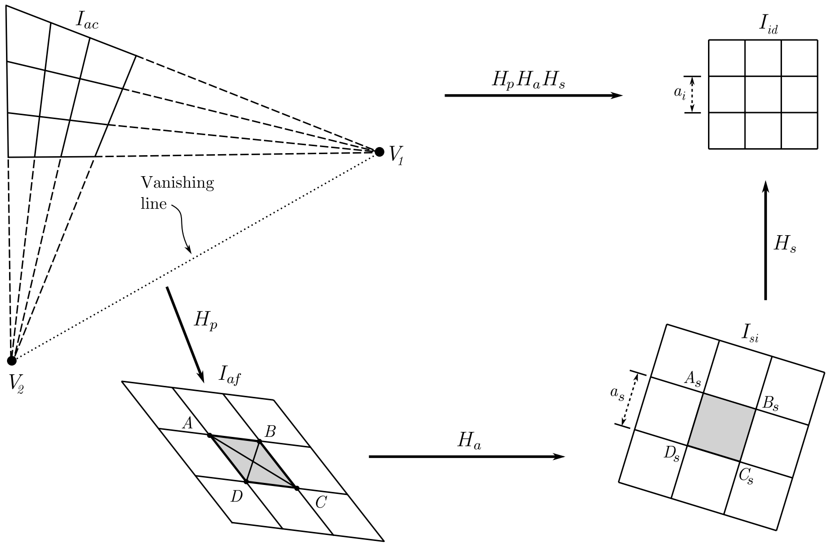


Fig. 6. Rectification process.

$$H_p = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ l_1 & l_2 & l_3 \end{pmatrix}, \tag{5}$$

where l_1, l_2, l_3 are the vanishing line parameters measured in I_{ac}.

Accordingly, the affine transformation matrix H_a is given by Eq. 6:

$$H_a = \begin{pmatrix} \frac{1}{\beta} & -\frac{\alpha}{\beta} & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \tag{6}$$

where parameters α and β are evaluated using the line segment coordinates in I_{af} grid and the length ratio of their corresponding segments in I_{si}.

In detail let $ABCD$ be a parallelogram in the grid of I_{af}, which is mapped to the square $A_sB_sC_sD_s$ after applying the affine transformation H_a shown in Fig. 6. Then α and β are given by the coordinates of the two intersection points of the circles with centers at (c_{x1}, c_{y1}), (c_{x2}, c_{y2}) and radii r_1, r_2 respectively, as given by Eqs. 7–10:

$$(c_{x1}, c_{y1}) = \left(\frac{x_{AB}y_{AB} - s_1^2 x_{BC}y_{BC}}{y_{AB}^2 - s_1^2 y_{BC}^2}, 0 \right) = \left(\frac{x_{AB}y_{AB} - x_{BC}y_{BC}}{y_{AB}^2 - y_{BC}^2}, 0 \right), \tag{7}$$

$$r_1 = \left| \frac{s_1(x_{BC}y_{AB} - x_{AB}y_{BC})}{y_{AB}^2 - s^2 y_{BC}^2} \right| = \left| \frac{(x_{BC}y_{AB} - x_{AB}y_{BC})}{y_{AB}^2 - y_{BC}^2} \right|, \tag{8}$$

and

$$(c_{x2}, c_{y2}) = \left(\frac{x_{AC}y_{AC} - s_2^2 x_{BD}y_{BD}}{y_{AC}^2 - s_2^2 y_{BD}^2}, 0 \right) = \left(\frac{x_{AC}y_{AC} - x_{BD}y_{BD}}{y_{AC}^2 - y_{BD}^2}, 0 \right) \tag{9}$$

$$r_2 = \left| \frac{s_2(x_{BD}y_{AC} - x_{AC}y_{BD})}{y_{AC}^2 - s^2 y_{BD}^2} \right| = \left| \frac{(x_{BD}y_{AC} - x_{AC}y_{BD})}{y_{AC}^2 - y_{BD}^2} \right|, \tag{10}$$

where $(x_{AB}, y_{AB})^T = \overrightarrow{AB}$, $(x_{BC}, y_{BC})^T = \overrightarrow{BC}$, $(x_{AC}, y_{AC})^T = \overrightarrow{AC}$, $(x_{BD}, y_{BD})^T = \overrightarrow{BD}$, $s_1 = \frac{A_sB_s}{B_sC_s} = 1$, and $s_2 = \frac{A_sC_s}{B_sD_s} = 1$.

Since the y coordinate of the center of both circles is zero, it yields out that the two intersection points have the same α and two opposite β values given by Eq. (11), (12):

$$\alpha = \frac{c_{x2} + c_{x1}}{2} + \frac{(r_1^2 - r_2^2)}{2(c_{x2} - c_{x1})} \tag{11}$$

$$\beta = \pm \frac{1}{2(c_{x2} - c_{x1})} \times \sqrt{[(r_1 + r_2)^2 - (c_{x2} - c_{x1})^2][(c_{x2} - c_{x1})^2 - (r_2 - r_1)^2]}. \tag{12}$$

Matrix H_a is formed using only the non negative β value [14], which leads to a simpler implementation.

Finally matrix H_s, which defines a similarity transformation between I_{si} and I_{id}, is given by Eq. 13:

$$H_s = \begin{pmatrix} & R & x_0 \\ & & y_0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} c & 0 & 0 \\ 0 & c & 0 \\ 0 & 0 & 1 \end{pmatrix}, \tag{13}$$

where c is a scaling factor, $(x_0, y_0)^T$ is a translation vector, and R is the 2×2 rotational matrix

$$R = \begin{pmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{pmatrix}. \tag{14}$$

The scaling parameter c is computed as $c = a_i/a_s$, where a_s, a_i are the EI edge sizes as mapped on I_{si} and I_{id}, respectively, as shown in Fig. 6. It should be noted that c is needed only in the case where I_{si} has to be scaled to a target display device. Hence a_i can be calculated by using the LA and the LCD pitch sizes. In a case where no scaling is required, the scaling factor is $c = 1$.

Parameter θ is evaluated in such a way that the I_{si} grid lines align to the grid lines in I_{id}. For the purposes of our work, vector $(x_0, y_0)^T$ is set equal to zero.

B. Implementation Details

In most practical cases [6] the above mathematical model cannot be directly applied. In detail, certain augmentations are needed to compensate for the errors introduced during the line segment registration stage. These errors cause deviations in the values of the ideal parameters of the grid lines in I_{ac}, which consequently converge within a small area rather than intersect at a single point.

Moreover the errors in the coordinates of the ideal vanishing points V_1 and V_2 will be propagated in parameters l_1, l_2, l_3 and will affect the grid in I_{af}. Eventually parallelograms in I_{af} are transformed into convex quadrilaterals with strong size fluctuations due to the fact that not all the grid lines are registered.

In order to compensate for the above issues we calculate the intersection point for each pair of concurrent lines and the *median of all intersection points* is taken as the vanishing point of the line set. The use of the median values ensures that outliers have no effect over the calculation of the vanishing point coordinates and hence increases the robustness of the proposed solution. The coordinates of the two estimated vanishing points are used to produce the vanishing line in the form given by Eq. 15:

$$l_1 \cdot x + l_2 \cdot y + l_3 = 0, \tag{15}$$

where $l_1, l_2, l_3 \in \mathbb{R}$ are used for constructing H_p. The resulting transformation matrix H_p is applied on the two line equation sets L_h and L_ν, producing L_{hn} and $L_{\nu n}$.

In the ideal case each of the derived line sets L_{hn} and $L_{\nu n}$ should contain lines with identical slopes and sequential interline distances and should construct a perfect parallelogram grid, as shown in I_{af} in Fig. 6. Any parallelogram of this grid can be used to compute the transformation matrix H_a given by Eq. (6). However in practical cases small deviations in slope as well as in distance values result in a grid that does not consist of perfect parallelograms. Besides some lines have failed to be identified so the grid is neither complete. For these reasons we estimate the *mean affine EI* and this is used for the computation of H_a. For the mean grid parallelogram estimation we first apply the following steps to each of the L_{hn} and $L_{\nu n}$ line sets:

i. We sort the lines according to their *x–intercept* value for L_{hn} or *y–intercept* value for $L_{\nu n}$.

ii. We calculate the distance between sequential lines in the set and construct a sequence of interline distances denoted as $\{d_n\}$. Since the lines of the set are not precisely parallel, for each line pair the mean value of its slope is used as the slope of both lines for this evaluation.

iii. We calculate the median value of sequence $\{d_n\}$ and denote it as d_h or d_ν accordingly.

iv. We calculate the median value of the slope of the line set and we denote it as s_h or s_ν accordingly.

Finally we use two pairs of parallel lines with distances d_h and d_ν and slopes s_h and s_ν, as shown in Fig. 7. The intersections of these lines define the mean parallelogram *ABCD*, which corresponds to *ABCD* in I_{af} in Fig. 6. In order to calculate the affine matrix H_a defined by Eq. (6) we first define $\phi_h = \arctan(s_h)$, $\phi_\nu = \arctan(s_\nu)$ and ϕ as shown in Fig. 7.

Subsequently we derive AB and BC as

$$AB = \frac{d_\nu}{\sin\phi} \quad \text{and} \quad BC = \frac{d_h}{\sin\phi}. \tag{16}$$

Next we consider that $\overrightarrow{AB} = (x_{AB}, y_{AB})^T$ and $\overrightarrow{BC} = (x_{BC}, y_{BC})^T$ and deduce Eqs. (17–20):

$$x_{AB} = AB\cos\phi_h = \frac{d_\nu}{\sin\phi}\cos\phi_h \tag{17}$$

$$y_{AB} = AB\sin\phi_h = \frac{d_\nu}{\sin\phi}\sin\phi_h \tag{18}$$

$$x_{BC} = BC\cos\phi_\nu = \frac{d_h}{\sin\phi}\cos\phi_\nu \tag{19}$$

$$y_{BC} = BC\sin\phi_\nu = \frac{d_h}{\sin\phi}\sin\phi_\nu. \tag{20}$$

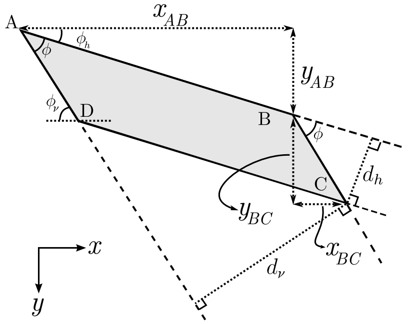


Fig. 7. Mean affine EI.

Substituting Equations (17–20) in Eq. (7) and Eq. (8), we conclude that:

$$(c_{x1}, c_{y1}) = \left(\frac{d_\nu^2 \cos\phi_h \sin\phi_h - d_h^2 \cos\phi_\nu \sin\phi_\nu}{d_\nu^2 \sin^2\phi_h - d_h^2 \sin^2\phi_\nu}, 0\right)$$
$$= \left(\frac{d_\nu^2 \sin 2\phi_h - d_h^2 \sin 2\phi_\nu}{d_\nu^2(1 - \cos 2\phi_h) - d_h^2(1 - \cos 2\phi_\nu)}, 0\right) \quad (21)$$

and

$$r_1 = \left|\frac{d_\nu d_h \cos\phi_\nu \sin\phi_h - d_h d_\nu \cos\phi_h \sin\phi_\nu}{(d_h \sin\phi_h)^2 - (d_\nu \sin\phi_\nu)^2}\right|$$
$$= \frac{d_\nu d_h |\sin(\phi_h - \phi_\nu)|}{|(d_h^2(1 - \cos 2\phi_h) - d_\nu^2(1 - \cos 2\phi_\nu)|}. \quad (22)$$

Finally substituting all trigonometric functions of Eq. (21) and Eq. (22) as functions of $\tan\phi_h = s_h$ and $\tan\phi_\nu = s_\nu$, we derive:

$$(c_{x1}, c_{y1}) = \left(\frac{d_\nu^2 s_h(1 + s_\nu^2) - d_h^2 s_\nu(1 + s_h^2)}{d_\nu^2 s_h^2(1 + s_\nu^2) - d_h^2 s_\nu^2(1 + s_h^2)}, 0\right) \quad (23)$$

$$r_1 = \frac{d_h d_\nu \sqrt{(1 + s_\nu^2)(1 + s_h^2)(s_\nu - s_h)^2}}{|d_\nu^2 s_h^2(1 + s_\nu^2) - d_h^2 s_\nu^2(1 + s_h^2)|}. \quad (24)$$

In addition, from Fig. 7 we observe that $\overrightarrow{AC} = \overrightarrow{AB} + \overrightarrow{BC}$ and $\overrightarrow{BD} = \overrightarrow{BC} - \overrightarrow{AB}$, so $x_{AC} = x_{AB} + x_{BC}$, $y_{AC} = y_{AB} + y_{BC}$, $x_{BD} = x_{BC} - x_{AB}$, and $y_{BD} = y_{BC} - y_{AB}$. In this way, substituting Eqs. (17–20) in Eq. (9) and Eq. (10) and working in a similar way as for Eqs. (21) and (22) we deduce that:

$$(c_{x2}, c_{y2}) = \left(\frac{s_\nu + s_h}{2 s_\nu s_h}, 0\right) \quad (25)$$

and

$$r_2 = \frac{|s_\nu - s_h|}{2 s_\nu s_h}. \quad (26)$$

Based on these calculations we derive the values for α and β, which are substituted in matrix H_a given by Eq. (6).

We finally evaluate the similarity transformation matrix H_s given by Eq. (10). In order to calculate the values for the rotational matrix R, which rotates I_{si} to align its sides to the sides of I_{id}, we define two unit vectors, $\hat{\upsilon}_{sh}$, and $\hat{\upsilon}_{s\nu}$, parallel to $\overrightarrow{A_sB_s}$ and $\overrightarrow{C_sB_s}$, respectively. The calculation for vectors $\hat{\upsilon}_{sh}$ and $\hat{\upsilon}_{s\nu}$ yields:

$$\hat{\upsilon}_{sh} = \frac{\overrightarrow{A_sB_s}}{\|\overrightarrow{A_sB_s}\|} = \frac{H_{\alpha\beta}(1, s_h)^T}{|H_{\alpha\beta}(1, s_h)^T|} = \frac{(1 - \alpha s_h, \beta s_h)^T}{\sqrt{(1 - \alpha s_h)^2 + \beta^2 s_h^2}} \quad (27)$$

and

$$\hat{\upsilon}_{s\nu} = \frac{(1 - \alpha s_\nu, \beta s_\nu)^T}{\sqrt{(1 - \alpha s_\nu)^2 + \beta^2 s_\nu^2}}. \quad (28)$$

Using $\hat{\upsilon}_{sh}$ and $\hat{\upsilon}_{s\nu}$ and the fact that $\hat{\upsilon}_{sh} \perp \hat{\upsilon}_{s\nu}$, we rewrite Eq. (14) in the form:

$$R = \begin{pmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{pmatrix} = \begin{pmatrix} \hat{\upsilon}_{sh}^T \\ \hat{\upsilon}_{s\nu}^T \end{pmatrix}^{-1} = \begin{pmatrix} \hat{\upsilon}_{sh}^T \\ \hat{\upsilon}_{s\nu}^T \end{pmatrix}^T$$
$$= (\hat{\upsilon}_{sh} \quad \hat{\upsilon}_{s\nu}). \quad (29)$$

In a case where no scaling is required, $c = 1$ and H_s takes the form:

$$H_s = \begin{pmatrix} (\hat{\upsilon}_{sh} & \hat{\upsilon}_{s\nu}) & \begin{matrix} 0 \\ 0 \end{matrix} \\ 0 \quad 0 & 1 \end{pmatrix}. \quad (30)$$

If scaling is required for a target display device, we calculate scaling factor c based on the values of a_i derived from the LA and the LCD pitch sizes and the value for a_s, which is the side length of $A_sB_sC_sD_s$. It is evident from Fig. 6 that a_s can be written as:

$$a_s^2 = \|\overrightarrow{A_sB_s}\|^2 = \|H_{\alpha\beta}\overrightarrow{AB}\|^2, \quad (31)$$

where $H_{\alpha\beta}$ is given by Eq. (32):

$$H_{\alpha\beta} = \begin{bmatrix} 1/\beta & -\alpha/\beta \\ 0 & 1 \end{bmatrix}. \quad (32)$$

Substituting $H_{\alpha\beta}$ and $\overrightarrow{A_sB_s} = (x_{AB}, y_{AB})^T$ in Eq. (31) we derive a_s:

$$a_s^2 = \frac{d_\nu^2(s_\nu^2 + 1)[(1 - \alpha s_h)^2 + \beta^2 s_h^2]}{\beta^2(s_\nu - s_h)^2}. \quad (33)$$

Finally the complete equation for matrix H_s:

$$H_s = \begin{pmatrix} (\hat{v}_{sh} \quad \hat{v}_{s\nu}) & \begin{matrix} 0 \\ 0 \end{matrix} \\ 0 \quad 0 & 1 \end{pmatrix} \begin{pmatrix} c & 0 & 0 \\ 0 & c & 0 \\ 0 & 0 & 1 \end{pmatrix}. \tag{34}$$

Substituting H_p, H_a, and H_s in Eq. (4) we derive matrix H, which is used for applying the compound transformation on I_{ac} to produce an estimate of I_{id} denoted as $\hat{I}_{\mathrm{id}}$.

Concluding this section, it should be noted that since not all the lines are detected and registered, a number of issues arise during further processing steps that require knowledge of the size and position of each EI. To this end we introduce a method for removing a small number of offending grid lines and reconstructing the missing ones in order to provide this additional information to later processing stages. In detail a small number of offending lines still exist after transforming the line equations L_h and L_ν from I_{ac} into $\hat{I}_{\mathrm{id}}$. In order to remove these lines we use their x or y intersect values, respectively, to deduce their interline distances. For each line we calculate its distances from all other parallel lines in the group in multiples of a_i and store these values in a sequence $\{D_i\}$. If more than half of the sequence $\{D_i\}$ members are equal to an integer within a tolerance value of $0.1 \cdot a_i$ the line is considered to be part of the grid. After identifying the grid lines, parallel lines are inserted in distances of a_i, hence reconstructing the missing lines of the grid. The reconstructed grid is shown in Fig. 8(g)–8(i).

4. Experiments and Results

In order to assess the performance of the proposed algorithm we calculated the error in the estimation of the rectification parameters l_1, l_2, α, β, and θ. Furthermore we evaluated the geometric consistency of the rectified InIms as described later in this section.

A. Test InIms

A large number of different InIms were generated using the ray tracing technique described in [15] with

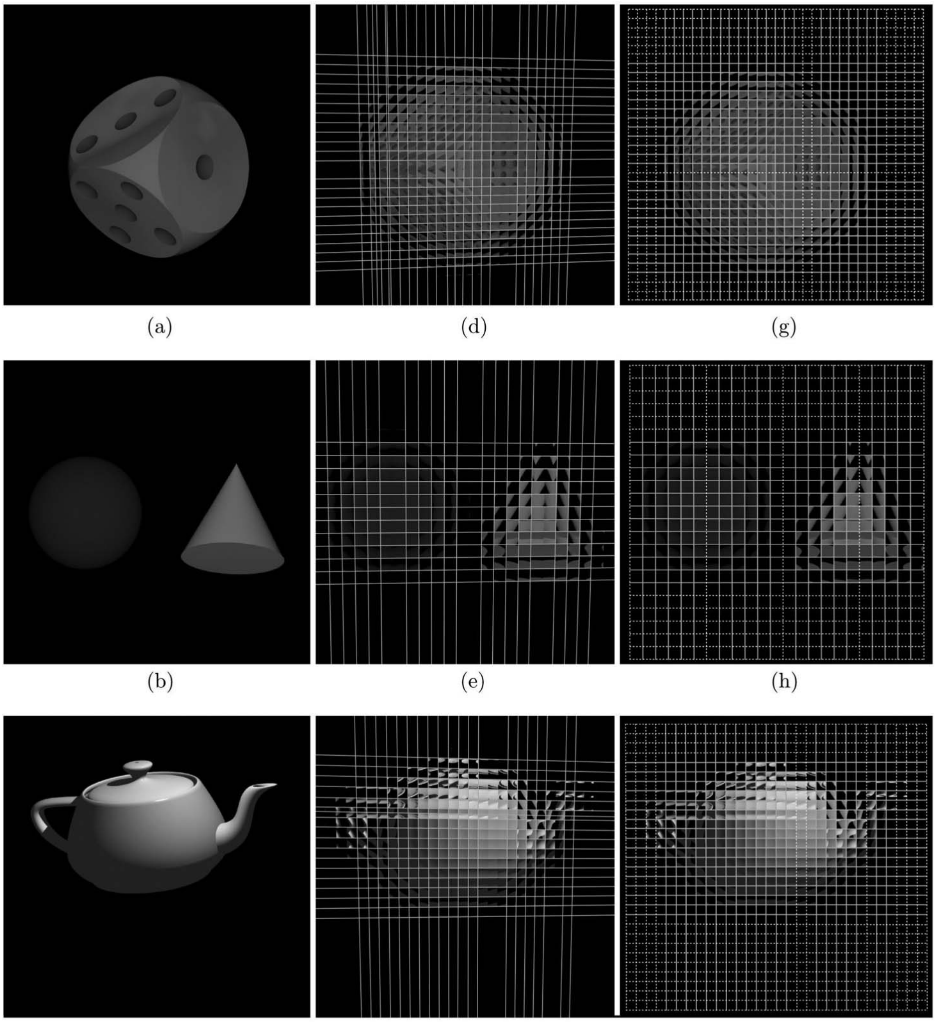


Fig. 8. Three characteristic InIms. (a)-(c) 2D rendered images of three 3D objects. (d)-(f) Acquired InIms with the registered grid superimposed. (g)-(i) Rectified InIms along with the reconstructed grid.

different scenes, EI sizes, texture levels, and color gradients. Different noise levels and distortion angles were used for assessing the performance of the algorithm. The images were perspectively distorted with known distortion matrices; therefore their ideal rectification parameters are known. In addition three different noise levels were introduced in each of the InIms in order to objectively assess the quality of the rectified grid produced by our framework in the range of 20 dB–30 dB with respect to the noiseless InIm. This approach was followed as it provides complete control over the introduced InIm distortion and knowledge of the ideal rectification parameters to be used as ground-truth values. It should be noted that no objective lens distortions are accounted for in the computer generated InIm experiments.

B. Parameters Evaluation

As already stated, our framework estimates parameters l_1, l_2, α, β and θ, which are required for the rectification of the InIm. Parameters l_1, l_2 represent the vanishing line coordinates in homogeneous representation, while α and β are the metric correction parameters and θ is the final rotation angle.

In order to assess the geometric consistency of the rectified InIm, we introduce two additional evaluation parameters that can be measured on the rectified grid, namely the angles of the adjacent edges of the EIs and the edge ratio for each EI. In detail we measured:

1. All four internal angles for each EI. We form a sequence $\{\omega_i\}$ of the angles in all the EIs in the rectified grid.
2. The four side length ratios, formed by considering each pair of adjacent edges of each EI. We form a sequence $\{\lambda_i\}$ of the adjacent edge ratios in the rectified grid.

As all the InIms used in the experiments had a rectangular EI grid structure, the ground truth values for these parameters are $\omega_{\mathrm{id}} = 90°$ and $\lambda_{\mathrm{id}} = 1$. In this section we present the calculated mean value and standard deviation for all the angles and edge ratios in each InIm.

C. Results

The calculated parameters used in the rectification matrices were evaluated over the complete set of the test InIms described previously. Here we present the evaluation results over three characteristic scenes for which the 2D rendered images are shown in Figs. 8(a)–8(c). In Figs. 8(d)–8(f) we also show the acquired InIm along with the registered lines as they are calculated after the line segment clustering process. In Figs. 8g–8(i) we show the rectified InIms along with the reconstructed grid lines as described in Section 3.

For each of these InIms we considered different noise levels and calculated the absolute error percentages for the evaluated parameters used in the rectification process. The results for the three scenes are summarized in Tables 1–3. As shown in Tables 1–3 the error for all the parameters used in the rectification pipeline did not exceed 4% for the worst case scenario where the InIms had high noise levels that reduced the InIm quality to 20 dB. For medium noise levels and InIm quality around 25 dB no parameter exceeded a maximum absolute error of 3% while for low noise levels and InIm quality at 30 dB the maximum absolute error for all parameters did not exceed 1.2%. However it should be noted that since not all parameters have the same impact to the error they introduce in the final rectified image these percentages do not provide an detailed representation of the final rectification error.

For this purpose we evaluated the geometric consistency of the rectified grid lines, as already described, by assessing the deviation of the side length ratios and the internal angles of the rectified EIs. The results for these measures are shown for each image in Tables 4–6. The results reported in Tables 4–6 show that the calculated mean internal angle value $\bar{\omega}_i$ is 90° regardless of the noise levels. In addition the standard deviation value σ_{ω_i} in all

Table 1. Error Percentages for the Rectification Parameters of the InIm in Fig. 8(g)

	l_1	l_2	α	β	θ
noiseless	0.2295%	0.4590%	0.6771%	0.0112%	0.3817%
30 dB	0.4812%	0.0210%	0.5834%	0.0376%	0.5273%
25 dB	0.3966%	0.1782%	0.5795%	0.0170%	0.6057%
20 dB	1.1237%	3.9952%	1.0681%	0.0810%	1.1398%

Table 2. Error Percentages for the Rectification Parameters of the InIm in Fig. 8(h)

	l_1	l_2	α	β	θ
noiseless	0.3698%	1.0473%	1.8463%	0.0104%	0.1234%
30 dB	0.5664%	1.1408%	0.6657%	0.0158%	0.3419%
25 dB	0.3006%	1.3307%	0.4325%	0.0467%	0.7342%
20 dB	3.8497%	0.9162%	0.6294%	0.2491%	2.4567%

Table 3. Error Percentages for the Rectification Parameters of the InIm in Fig. 8(i)

	l_1	l_2	α	β	θ
noiseless	1.1759%	0.6608%	1.6784%	0.0123%	0.1374%
30 dB	0.3683%	1.1985%	0.5588%	0.1109%	0.3831%
25 dB	0.1038%	2.9553%	0.0585%	0.0456%	0.4179%
20 dB	1.3484%	1.9558%	1.7878%	0.3300%	0.4517%

Table 4. Geometric Consistency Evaluation Parameters for the InIm in Fig. 8(g)

	$\bar{\omega}_i \pm \sigma_{\omega_i}$	$\bar{\lambda}_i \pm \sigma_{\lambda_i}$
noiseless	$90.000° \pm 0.049°$	1.005 ± 0.010
30 dB	$90.000° \pm 0.050°$	0.999 ± 0.011
25 dB	$90.000° \pm 0.052°$	1.009 ± 0.012
20 dB	$90.000° \pm 0.070°$	1.010 ± 0.014

Table 5. Geometric Consistency Evaluation Parameters for the InIm in Fig. 8(h)

	$\bar{\omega}_i \pm \sigma_{\omega_i}$	$\bar{\lambda}_i \pm \sigma_{\lambda_i}$
noiseless	$90.000° \pm 0.097°$	1.0070 ± 0.0066
30 dB	$90.00° \pm 0.11°$	1.010 ± 0.010
25 dB	$90.00° \pm 0.19°$	0.999 ± 0.016
20 dB	$90.00° \pm 0.21°$	1.019 ± 0.019

Table 6. Geometric Consistency Evaluation Parameters for the InIm in Fig. 8(i)

	$\bar{\omega}_i \pm \sigma_{\omega_i}$	$\bar{\lambda}_i \pm \sigma_{\lambda_i}$
noiseless	$90.000° \pm 0.060°$	0.996 ± 0.011
30 dB	$90.00° \pm 0.10°$	1.009 ± 0.013
25 dB	$90.00° \pm 0.12°$	1.009 ± 0.017
20 dB	$90.00° \pm 0.16°$	0.999 ± 0.024

cases and for all different noise levels did not exceed 0.21°. The error trend for the mean integral angles is shown in Fig. 9. Finally the mean side length ratios $\bar{\lambda}_i$ for the EIs are in all cases within a very narrow range around the nominal value λ_{id} with a small standard deviation σ_{λ_i}, which in the worst case scenario is $24 \cdot 10^{-3}$. The error trend for the mean side length ratio is shown in Fig. 10.

4.D. Validation with Optically Acquired InIms

An additional set of optically acquired InIms was also used to demonstrate the robustness of the proposed method over real world data. For the optically acquired InIms, we adopted the pickup process used in [8] with the same distortion parameter settings.

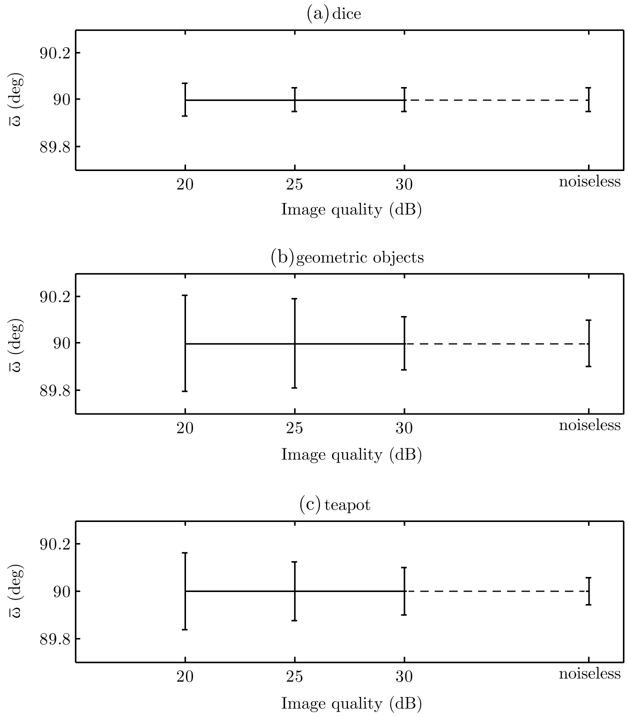


Fig. 9. Mean internal angles versus image quality.

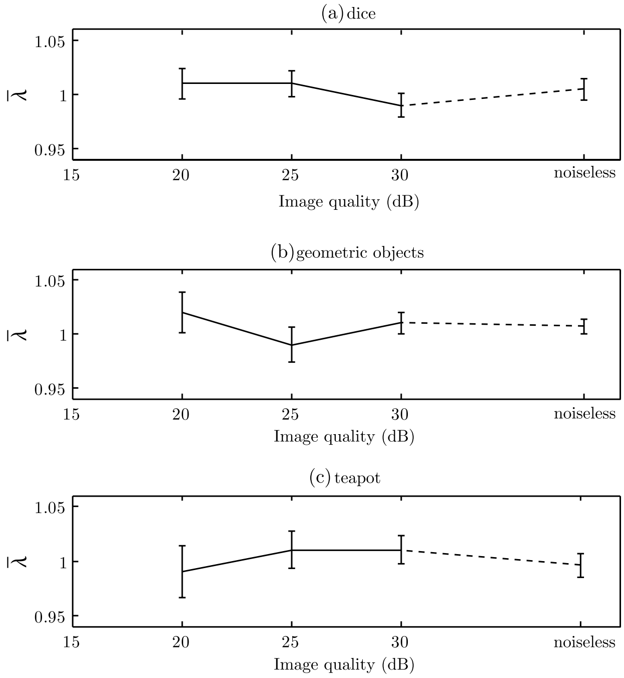


Fig. 10. Mean side length ratio versus image quality.

Pincushion and barrel distortions were corrected using the camera firmware of Canon EOS-500D, which was used in our case and are not considered for correction in the presented rectification framework. It should be noted that nowadays once the characteristics of the camera objective lens and the camera type are known, the barrel and pincushion distortions can be corrected with a variety of software toolkits as already reported in [8]. Here we present the results for an optically acquired InIm of a dice shown in Fig. 11(a). The acquired InIm with the superimposed grid is shown in Fig. 11(b) while the rectified InIm with the reconstructed grid is shown in Fig. 11(c). The parameter values summarized in Tables 7–8 show that the results of the algorithm for optically acquired InIms are in line with the results in the computer generated InIms.

5. Conclusions

In this paper we presented a framework that performs InIm rectification in order to compensate for a number of situations in an InIm acquisition device where the LA and the CCD planes are not parallel.

One of the strong features of the proposed framework is the minimal set of required parameters that should be entered in the system beforehand. More precisely the required parameters are the LA and LCD pitch sizes. It should be noted that these parameters are needed only in the case where scaling of the rectified InIm to a display device is necessary. The accuracy of the proposed framework is validated through the geometric consistency of the rectified InIm grids over a set of computer generated and optically acquired InIms with different complexity, texture, and noise characteristics. The results show

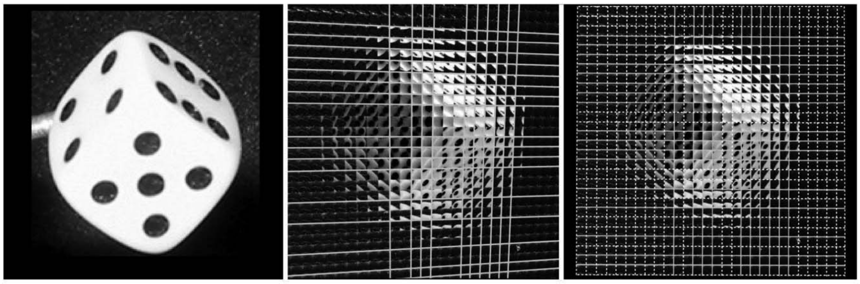

(a) (b) (c)

Fig. 11. Optically acquired dice. (a) Photograph of the dice. (b) Acquired InIm with the registered grid superimposed. (c) Rectified InIm along with the reconstructed grid.

only small deviations from the ideal internal angles values while their mean value is always equal to ω_{id}. Specifically our framework differentiates from previously proposed solutions [8] as it utilizes all detected line segments of the InIm to perform the rectification of the InIm grid. The large number of the InIm grid line segments that are detected through the use of the LSD algorithm provides strong statistical support to the registration of grid lines through the clustering process. This statistical approach greatly reduces the number of outliers and practically minimizes the introduced error in later stages of the rectification framework. The accuracy of our framework is further increased through the method used for determining the vanishing points. Using more than one pair of concurrent lines our framework reduces the error in the localization of the vanishing points. It should be noted that small errors at the vanishing point detection process may result in poor rectification of the InIm grid as the determination of these points affect the whole InIm.

The robustness of the proposed framework is also validated through the fact that the accuracy of rectified grids are almost invariant despite the use of different scenes as well as different noise characteristics. The high fault tolerance of our framework is also attributed to the statistical nature of the procedures utilized for determining the line equations and the vanishing points. The use of the LSD for the LSD and registration process and the introduced metric for the determination of collinear segments further enhance the robustness of the proposed framework by ruling out most of the non grid line segments.

Table 7. Error Percentages of the Rectification Parameters for the InIm in Fig. 11

l_1	l_2	α	β	θ
0.3267%	0.0264%	0.564%	0.0204%	0.324%

Table 8. Geometric Consistency Parameters for the InIm in Fig. 11

$\bar{\omega}_i \pm \sigma_{\omega_i}$	$\bar{\lambda}_i \pm \sigma_{\lambda_i}$
$90.000° \pm 0.055°$	1.006 ± 0.010

Moreover the construction and application of a single transformation matrix H instead of the sequential application of the intermediate matrices of Eq. (4) greatly reduces the time needed for rectifying an InIm. Additional time performance gain is achieved through the use of low complexity algorithms like the LSD which is used for the LSD and registration process. The proposed framework constitutes a valuable tool for all InIm processing tasks, providing accurate and robust rectification results without any user interaction.

This research has been cofinanced by the European Union (European Social Fund) and Greek national funds through the Operational Program "Education and Lifelong Learning" of the National Strategic Reference Framework (NSRF)—Research Funding Program: Heracleitus II. "Investing in knowledge society through the European Social Fund."

The authors would like to thank the reviewers for their accurate and constructive comments.

References

1. G. Lippmann, "La Photographie Integràle," Comptes-Rendus, Acad. Sci. **146**, 446–451 (1908).
2. J.-Y. Son, and B. Javidi, "Three-dimensional imaging methods based on multiview images," J. Disp. Technol. **1**, 125–140 (2005).
3. J. Park, Y. Kim, J. Kim, S. Min, and B. Lee, "Three-dimensional display scheme based on integral imaging with three-dimensional information processing," Opt. Express **12**, 6020–6032 (2004).
4. J. S. Jang, and B. Javidi, "Formation of orthoscopic three dimensional real images in direct pickup one-step integral imaging," Opt. Eng. **42**, 1869–1870 (2003).
5. G. Passalis, N. Sgouros, S. Athineos, and T. Theoharis, "Enhanced reconstruction of three-dimensional shape and texture from integral photography images," Appl. Opt. **46**, 5311–5320 (2007).
6. R. Hartley, and A. Zisserman, *Multiple View Geometry in Computer Vision*, 2nd ed. (Cambridge University, 2000).
7. N. Sgouros, S. Athineos, M. Sangriotis, P. Papageorgas, and N. Theofanous, "Accurate lattice extraction in integral images," Opt. Express **14**, 10403–10409 (2006).
8. K. Hong, J. Hong, J. Jung, J. Park, and B. Lee, "Rectification of elemental image set and extraction of lens lattice by projective image transformation in integral imaging," Opt. Express **18**, 12002–12016 (2010).

9. R. von Gioi, J. Jakubowicz, J.-M. Morel, and G. Randall, "LSD: A fast line segment detector with a false detection control," IEEE Trans. Pattern Anal. Machine Intell. **32**, 722–732 (2010).
10. A. Desolneux, L. Moisan, and J.-M. Morel, "Meaningful alignments," International Journal of Computer Vision **40**, 7–23 (2000).
11. M. Unser, and M. Eden, "Multiresolution feature extraction and selection for texture segmentation," IEEE Trans. Pattern Anal. Machine Intell. **11**, 717–728 (1989).
12. S. Theodoridis, and K. Koutroumbas, *Pattern Recognition*, 3rd ed. (Academic, 2006).
13. R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*, 2nd ed. (Wiley-Interscience, 2001).
14. D. Liebowitz, and A. Zisserman, "Metric rectification for perspective images of planes," in Proceedings 1998 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (IEEE Computer Society, 1998), pp. 482–488.
15. S. S. Athineos, N. P. Sgouros, P. G. Papageorgas, D. E. Maroulis, M. S. Sangriotis, and N. G. Theofanous, "Photorealistic integral photography using a ray-traced model of capturing optics," J. Electron. Imaging **15**, 043007 (2006).