Construction and detection of straight lines, distances, and circles in log-polar images

Konrad Schindler

Computer Graphics and Vision, TU Graz
Inffeldgasse 16, 8010 Graz, Austria
schindl@icg.tu-graz.ac.at

Abstract. This paper investigates, how two fundamental measurement operations in Euclidean space, namely the construction of a straight line of given orientation angle and the measurement of distances along this line, are mapped to the non-Euclidean log-polar image plane. It is shown, how these two operations can be efficiently implemented for discrete log-polar images. The measurement operations are used to perform line and circle detection in log-polar sampled images, a task, which can only be solved with the usual methods known from Cartesian images, if the two operations are available: chains of edgels are extracted from an image, then lines and circles are found through geometric constructions and testing of geometric relations. Experiments with both synthetic and real images are presented, and the detection results are quantitatively evaluated.

1 Introduction

This paper addresses the detection of geometric features in log-polar sampled images. The log-polar mapping is an imaging method with space-variant resolution, which has been introduced theoretically by Weiman and Chalkin [1] and by Sandini and Tagliasco [2] in an effort to mimic the imaging geometry of the human retina. The first sensor prototype was presented by Tistarelli and Sandini [3], which was later followed by an advanced version [4]. The main advantages of log-polar imaging are that it provides a large field of view on one hand and a high resolution in the image center on the other hand, while keeping the amount of data to be stored and processed low. An example showing the same image in Cartesian and in log-polar coordinates is depicted in Figure 1.

Besides the attention-sensitive data reduction, the log-polar geometry has also shown to have advantages for several robotics applications such as linear flow estimation [5], looming detection [6] and vergence control [7]. An important property of the log-polar image plane is that local operations such as gradient-based interest point detection can be performed in the traditional way, without the need for modifications.

On the other hand, however, many image computations are more complicated – for example, the stereo problem cannot be solved with conventional methods [8, 9], and the same is true for the detection of line features. The reason behind these difficulties is that the basic operations of Euclidean geometry commonly used to explore the image space, namely the measurement of Euclidean distances and directions, cannot be directly performed in log-polar coordinates, because the mapping is non-linear and not isotropic: straight lines are not mapped to straight lines, and the distance between neighboring pixels depends on their location in the image. In order to overcome the problem, this paper takes a closer look at the two fundamental operations of image measurement, namely the construction of straight lines of a given orientation and the measurement of distances along these lines, shows how they are mapped to the log-polar plane, and presents efficient methods for their implementation.

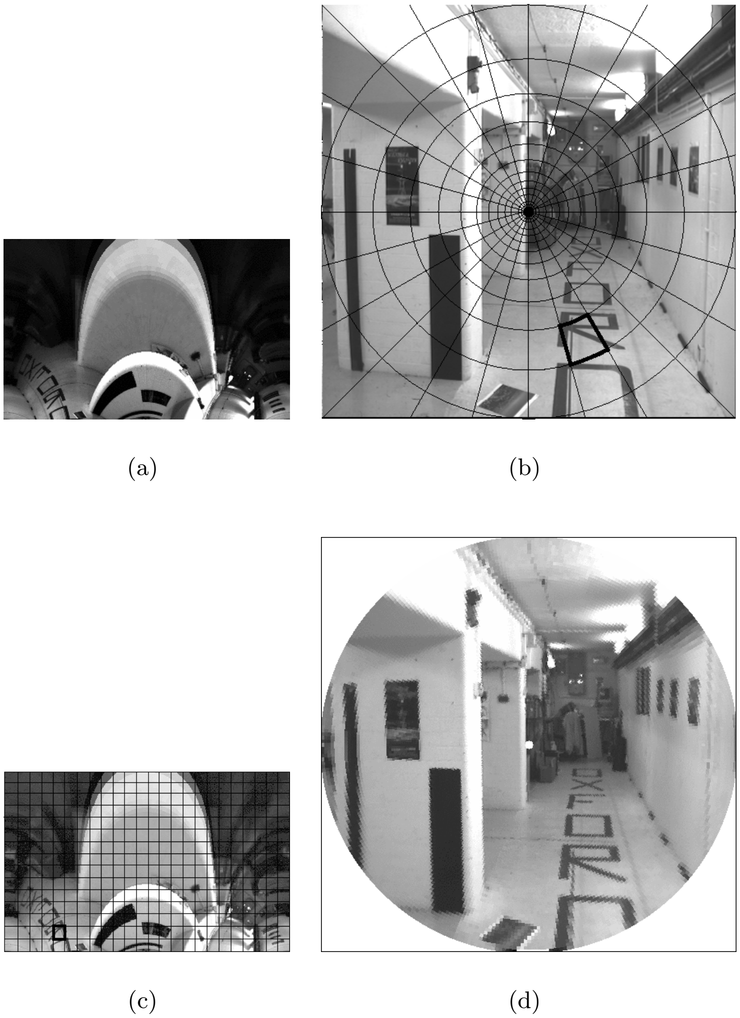
The target application is the detection of straight line segments and circles in log-polar sampled images. This problem has been addressed before in a clever way by Young [10]. The fundamental idea of his work is to convolve the image with a log-polar image of the searched feature, which has the same size. The method however involves non-linear preprocessing, and it is only able to detect features, which produce a significant peak of the convolution output, i.e., globally dominant features.

In contrast, we propose a more conventional approach to feature detection. Our work-flow starts with the detection of local edge elements ('edgels') with a gradient-based operator. Then edgels which are close to each other are iteratively linked to edgel chains, and these chains are inspected with a regression algorithm such as RANSAC in order to find straight line segments and circles. For the regression, we must on one hand be able to construct lines and circles in the log-polar plane, and on the other hand we must be able to test, whether edgels satisfy some metric constraints (e.g., whether their distance to some hypothetical circle center is equal to the circle's radius). In this part we will make use of the geometric operations in the log-polar plane.

The paper is structured as follows: in section 2, the basic properties of log-polar images are briefly reviewed, and the notation is introduced. In section 3, geometric properties of straight lines in log-polar coordinates are exploited to derive incremental algorithms for the two fundamental measurement operations. Section 4 presents feature detection results on both synthetic images with known ground truth and real images. In section 5 the paper is summarized and an outlook is given on potential future work.

2 Log-polar imaging

A log-polar image is produced through a projection onto an image plane, which is not sampled in a rectangular (x, y)-grid, but in the following way: the pixels are arranged in concentric circular rings around the focus of attention (in existing perspective log-polar cameras this is the principal point). On each ring the same number of pixels is sampled, and the pixel size increases exponentially with growing radius of the rings in such a way, that all pixels are approximately square (see Figure 1). An approximation of this sampling scheme can also be found in the ganglion cells of the retina and visual cortex of primates [11, 12].



(a) (b)

(c) (d)

Fig. 1. Log-polar imaging. (a) A log-polar image generated from the 'corridor' dataset with parameters $N = 360$ and $r_{min} = 95$. (b) The original image overlayed with a log-polar grid. (c) The log-polar image with the same grid overlayed. One grid cell has been marked in to illustrate the mapping procedure. (d) The log-polar image mapped back to Cartesian coordinates. Note the decrease in resolution with increasing radius. The size of a pixel is the same in all images to illustrate the data reduction.

Let the origin of the image coordinate system be the image center. Then points $\mathbf{x}_c = [x, y]^\mathsf{T}$ of the Cartesian image plane are transformed to points $\mathbf{x} = [r, \alpha]^\mathsf{T}$ in the log-polar plane by the following function, which is sometimes called the *logmap* λ:

$$\mathbf{x} = \lambda_U(\mathbf{x}_c) = \begin{bmatrix} \frac{1}{2}\log_U(x^2+y^2) \\ \arctan\frac{y}{x} \end{bmatrix} \tag{1}$$

The transformation is conformal, has a singularity in the coordinate center (which is overcome in practice by stopping at a minimal radius r_{min}), and it has only one parameter, namely the base of the logarithm U, which depends on the desired number N of pixels per ring via $U = \frac{N+2\pi}{N}$. In the images resulting from the log-polar sampling method, image rotations are mapped to shifts of the polar angle α, and expansions are mapped to shifts of r. The inverse mapping μ from log-polar to Cartesian coordinates (the *inverse logmap*) is given by

$$\mathbf{x}_c = \mu_U(\mathbf{x}) = \begin{bmatrix} x \\ y \end{bmatrix} = U^r \begin{bmatrix} \cos\alpha \\ \sin\alpha \end{bmatrix} \tag{2}$$

Log-polar images are commonly stored and displayed by plotting the (r, α)-values into an conventional Cartesian coordinate system with orthogonal axes, which generates an image with N rows and $(r_{max} - r_{min})$ columns. In this paper the axes of the log-polar images have been flipped for display purposes, so that the α-axis runs from the left to the right and the r-axis from top to bottom of the page.

3 Lines in the log-polar plane

The analytic expression for a straight line $\mathcal{L}$ in log-polar coordinates is given by

$$r = r_0 - \log\cos(\alpha - \alpha_0) \tag{3}$$

In this paper, we will however not use this expression directly, but rather parameterize a line through a point $\mathbf{p}_1$ and the tangent direction γ_1 of the line in $\mathbf{p}_1$. A finite line segment is either given by $\mathbf{p}_1$, γ_1 and a length l, or by two points $\mathbf{p}_1 = [r_1, \alpha_1]^\mathsf{T}$ and $\mathbf{p}_2 = [r_2, \alpha_2]^\mathsf{T}$.

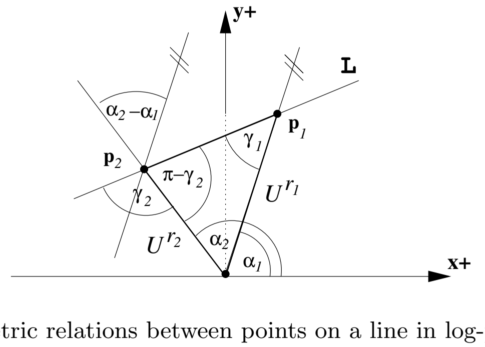


Fig. 2. Trigonometric relations between points on a line in log-polar coordinates.

Let us now consider the relation between two points $\mathbf{p}_1$, $\mathbf{p}_2$ on the line and the corresponding tangent directions γ_1, γ_2. Lines $\alpha_i = const.$ in the log-polar system correspond to radial lines of constant polar angle in the Cartesian plane (see Figure 2), which directly leads to the fundamental relation

$$\gamma_2 - \gamma_1 = \alpha_2 - \alpha_1 = \Delta\alpha \tag{4}$$

This expression states that for any point on a given line, α determines γ and vice versa. Elementary trigonometry gives the equation which is needed to construct a new point $\mathbf{p}_2$ on $\mathcal{L}$ given r_2:

$$\sin\gamma_2 = U^{r_2 - r_1}\sin\gamma_1 = U^{\Delta r}\sin\gamma_1 \tag{5}$$

Conversely, a point is constructed for a given polar angle α_2 through

$$r_2 = r_1 + \log_U \frac{\sin\gamma_2}{\sin\gamma_1} \tag{6}$$

If the straight line is given by two points rather than a point and a tangent, we have to compute the tangents first. For this task, γ_2 in equation (5) has to be substituted using equation (4), and after some straight-forward algebraic manipulations we get the expression

$$\tan\gamma_1 = \frac{\sin\Delta\alpha}{U^{-\Delta r} - \cos\Delta\alpha} \tag{7}$$

3.1 Constructing lines

In a log-polar image the polar angle α is uniformly sampled at discrete intervals $\alpha_S = \alpha_{i+1} - \alpha_i$. For neighboring pixels equation (4) therefore can be rewritten as

$$\gamma_{i+1} = \gamma_i + \alpha_S \tag{8}$$

and equation (6) becomes

$$r_{i+1} = r_i + \log_U \frac{\sin(\gamma_i + \alpha_S)}{\sin\gamma_i} = r_i + \Lambda(\gamma_i) \tag{9}$$

The function $\Lambda(x)$ has to be evaluated only for equally spaced discrete values in the range $0 \leq x_i \leq 2\pi$. It depends only on the tangent angle γ_i (except for the sensor parameters U and α_S) but *not* on the line $\mathcal{L}$. It can therefore be stored in a single, one-dimensional, sensor-specific lookup-table. With equations (8) and (9) we can incrementally construct a straight line in the image with only two additions per step by iteratively updating r and γ. Examples are shown in Figure 4.

3.2 Measuring distances

The second problem is how to measure distances in the log-polar plane, given that the scale of the mapping is space-variant, i.e., the same distance in the log-polar plane corresponds to different Euclidean distances in the Cartesian plane depending on the radius U^r from the coordinate center. In a discrete log-polar image we can regard the pixels as squares (the difference between the inner and outer side of a pixel is $< 3.5\%$ for $N > 180$), and the size d of a pixel is given by

$$d = U^{r+1} - U^r \tag{10}$$

The distance d_i from point $\mathbf{p}_i$ to the following point $\mathbf{p}_{i+1}$ on line $\mathcal{L}$ is thus given by

$$d_i = \frac{U^{r_{i+1}} - U^{r_i}}{\sin \gamma_i} = \frac{\mathit{\Gamma}(i)}{\sin \gamma_i} \tag{11}$$

Again, the function $\mathit{\Gamma}(x)$ is a one-dimensional lookup-table, which depends only on the sensor, and also the sine of γ in the denominator is only needed for discrete values and can be stored in a lookup-table. With equation (11) we can approximately measure the length along $\mathcal{L}$ by adding the d_i while constructing the line. This again can be done with low computational effort, namely one division and two additions per step.

3.3 Subpixel accuracy

A line $\mathcal{L}$ constructed with equation (9) has an exact floating point value $r(\alpha_i)$ for each row α_i. It may however be necessary to interpolate new points between two such discrete locations $\mathbf{p}_1$ and $\mathbf{p}_2$, for example when intersecting two lines. For practical image processing tasks it is sufficient to interpolate linearly in spite of the logarithmic scale: the error one commits when linearly interpolating r in the log-polar image is the distance d_r between the image of $\mathcal{L}$ and the straight line joining $\mathbf{p}_1$ and $\mathbf{p}_2$, measured along the r-axis (see Figure 3a). Let us assume – without loss of generality – two neighboring points $\mathbf{p}_1$, $\mathbf{p}_2$ on a vertical line, i.e., $\alpha_0 = 0$. For an arbitrary intermediate point $\alpha = \alpha_1 + t\alpha_S$, where $0 \leq t \leq 1$, the distance d_r is given by

$$d_r = \log_U \frac{\cos(\alpha_1 + t\alpha_S)}{\cos \alpha_1} - t \log_U \frac{\cos(\alpha_1 + \alpha_S)}{\cos \alpha_1} \tag{12}$$

As can be seen from Figure 3b, this error is negligible except for values of α_1 close to $\pm\frac{\pi}{2}$. Such values are only reached if the perpendicular distance from $\mathcal{L}$ to the origin is small, and even then only close to the image borders. If subpixel computations shall be carried out in such cases (which is unlikely because of the low resolution close to the image borders), a viable method is to compute $\mathit{\Lambda}(x)$ not only for a spacing of α_S, but for a fraction $\frac{\alpha_S}{k}$, $k \in \{2, 3, \ldots\}$ in order to reduce the error.

3.4 Combining operations

Figure 4 shows the two described operations in log-polar coordinates and how they can be combined to construct more complex geometric entities:

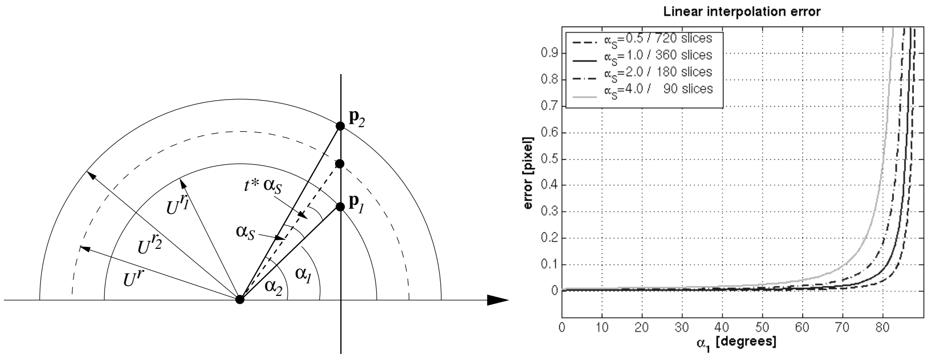


Fig. 3. Linear interpolation error of lines in log-polar coordinates. (a) Interpolation of a new point between neighboring rows. (b) The maximum interpolation error as a function of the point position α_1.

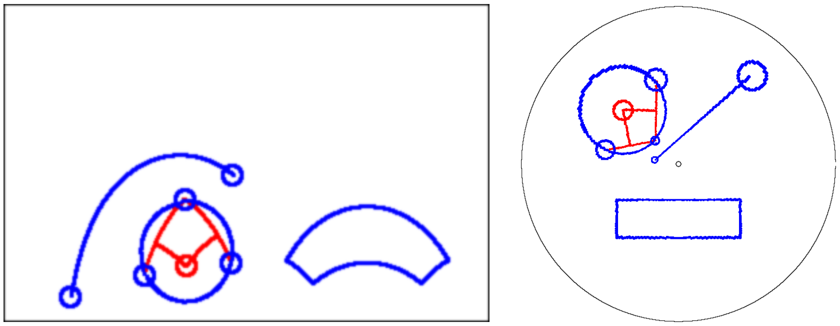
- A straight line segment between two given points is incrementally constructed as described above.
- A rectangle is constructed as a sequence of line segments of given length with a 90 degree turn at the end of each segment.
- The center of a circle through 3 points is constructed with the classical geometric construction: draw the straight line segment from the first to the second point and measure its length, find the midpoint of the segment, i.e., the point where the length is half of the total length, and construct a perpendicular line through the midpoint. Repeat the whole construction with the second and the third point, then intersect the two resulting lines.

4 Feature Detection

The principles described in the previous section have been used to implement conventional methods for line and circle detection in the log-polar domain. The following section describes the procedures used for feature detection, shows experimental results, and presents empirical results for the obtainable accuracies based on synthetic images with known ground truth. Special-purpose sensors with log-polar sampling have actually been built [4], but since no such sensor was available for the present research, the log-polar images were simulated by resampling conventional digital images to log-polar geometry.

4.1 Straight line detection

For line detection, edgel chains extracted at sub-pixel accuracy with the method first described by Canny [13] and refined by Rothwell [14] were extracted from



(a) (b)

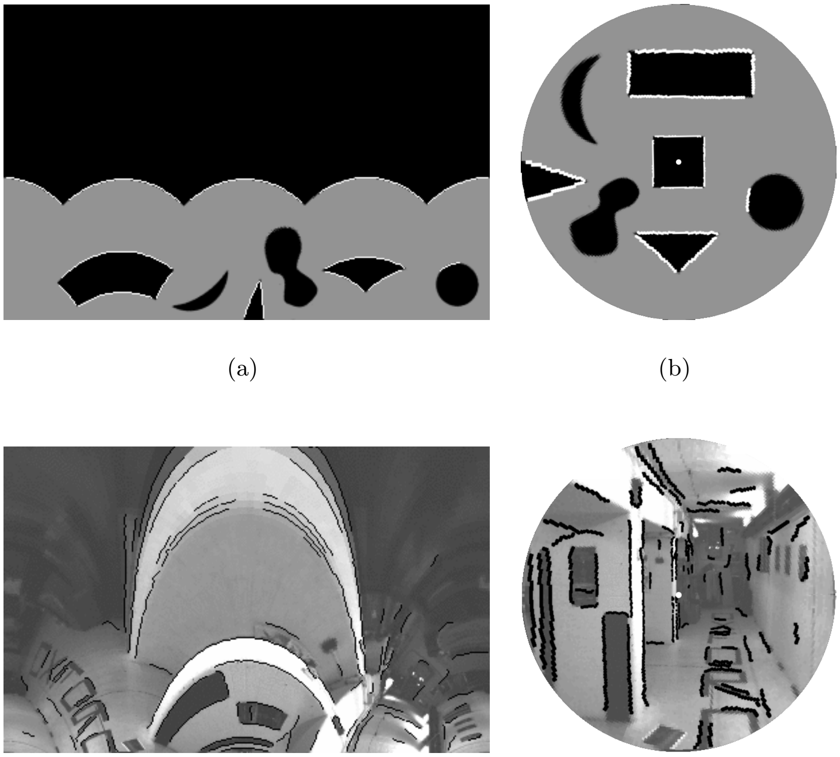
Fig. 4. Geometric constructions in log-polar coordinates. (a) Constructing a line segment between two points, a rectangle, and a circle. (b) Cartesian remapping of the image with constructed features. See text for details.

the images. For each point in a chain the local gradient was estimated, and a greedy search procedure was used to split the chain into subchains which are consistent with the tangent constraint from equation (4). Finally, subchains with few edgels were discarded with a simple threshold.

Figure 5 shows the results of line detection for two images. The first one is a synthetic image, which contains several linear structures, but also curved objects, in order to verify that the method can indeed discriminate between straight and curved lines. The image size is 360 × 234 pixels, corresponding to the log-polar parameters $U = 1.01745$, $r_{min} = 95$, $r_{max} = 328$. The second image is a log-polar remapping of one of the images of the 'corridor' sequence. The image size is 360 × 227 pixels with the parameters $U = 1.01745$, $r_{min} = 95$, $r_{max} = 321$. The first image and two more synthetic images with known line directions (all with the same parameters) were used to quantitatively evaluate the detected line sets. The results are displayed in Table 1.

	min	0.13°		*min*	0.01°		*min*	0.06°
	max	1.36°		*max*	2.06°		*max*	2.52°
	rms	0.80°		*rms*	0.78°		*rms*	1.07°
	DR	100%		*DR*	100%		*DR*	82.5%

Table 1. Accuracy of straight line detection in log-polar coordinates. The top 3 rows are the minimal, maximal and root mean square direction errors for each extracted line set. For the *rms*, each segment has been weighted with the number of contributing edgels. The last row is the detection rate *DR*. Segments are only missed close to the image border, where the resolution is low.



(a)

(b)

(c)

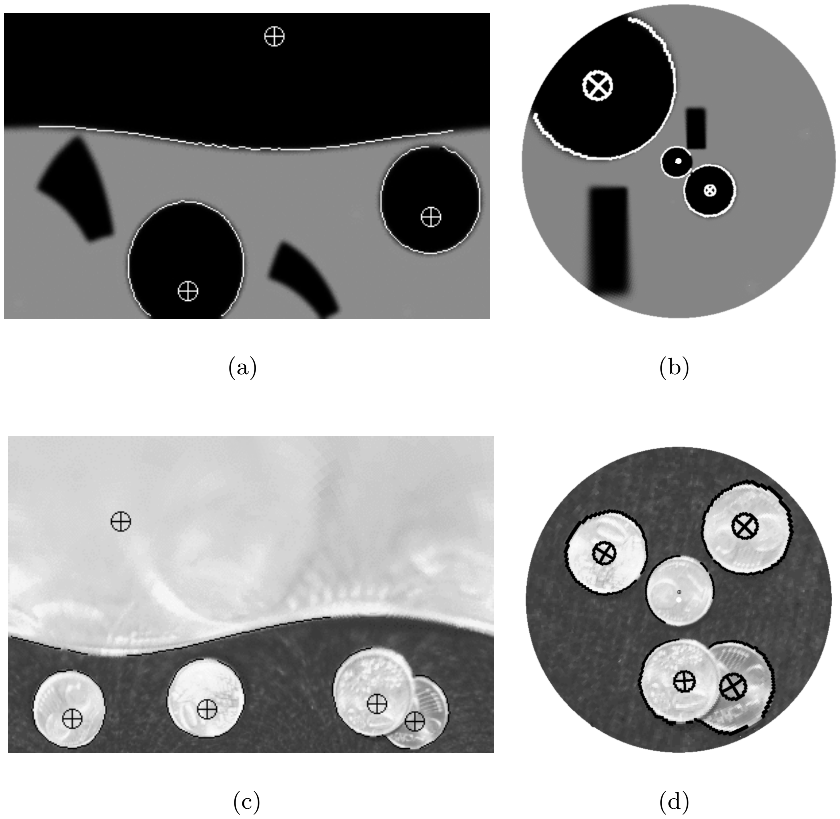
(d)

Fig. 5. Straight line detection in log-polar coordinates. (a),(b) Synthetic test image with detected lines in log-polar coordinates and remapped to Cartesian coordinates. (c),(d) 'corridor' image with detected lines in log-polar coordinates and remapped to Cartesian coordinates.

4.2 Circle detection

For circle detection the edgel chains were split at sharp creases, and a RANSAC scheme was set up to find a circle in each chain: Three distinct points were randomly drawn, the corresponding circle center was constructed as already shown in Figure 4, and the radius measured. Then a straight line segment was constructed from the center to each edgel in the chain and its length compared to the radius. This was repeated many times and only the best circle, i.e., the one with the largest number of consistent edgels, was retained for each chain. Finally, circles with too few edgels were again discarded with a simple threshold.

Detection results are displayed in Figure 6 and in Table 2. Again synthetic images were used to verify that the algorithm can correctly discriminate between circles and other structures, and to enable a quantitative evaluation. The log-polar images have 360×227 pixels, corresponding to the parameters $U = 1.01745$, $r_{min} = 95$, $r_{max} = 321$. The coordinates of the circle center were used to assess the accuracy. Since detection with RANSAC is a statistical process, it was repeated 20 times, and the root mean square error with respect to the known circle centers was used as error measure. The errors have been computed in both log-polar and Cartesian coordinates in order to demonstrate how the space-variant resolution affects the uncertainty. Note that the log-polar errors increase exponentially close to the center due to the exponential decrease in scale, whereas the Cartesian errors decrease due to the better resolution. As a real example, a picture of a few coins was taken with a digital camera and resampled to a log-polar image of 360×235 pixels with the parameters $U = 1.01745$, $r_{min} = 95$, $r_{max} = 329$.



(a) (b)

(c) (d)

Fig. 6. Circle detection in log-polar coordinates. (a),(b) Synthetic test image with detected circles in log-polar coordinates and remapped to Cartesian coordinates. (c),(d) 'coins' image with detected circles in log-polar coordinates and remapped to Cartesian coordinates.

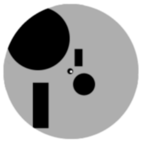
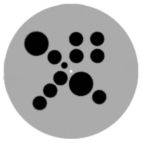
	circle	**lp[px]**	**ct[px]**		**circle**	**lp[px]**	**ct[px]**
	min_{lp}	1.8	5.5		min_{lp}	1.5	3.0
	max_{lp}	30.0	2.5		max_{lp}	3.5	3.8
	min_{ct}	30.0	2.5		min_{ct}	3.1	1.9
	max_{ct}	1.8	5.5		max_{ct}	3.3	9.8
	rms	17.4	3.8		rms	2.8	5.5

Table 2. Accuracy of circle detection in log-polar coordinates. 20 runs of RANSAC detection were performed on each image. The location error of the center is given in log-polar and Cartesian coordinates for the circles with the smallest and largest rms-error in log-polar coordinates (rows 1+2), and for those with the smallest and largest rms-error in Cartesian coordinates (rows 3+4). The last row is the rms-error calculated over all circles. In the first image, all 3 circles where detected, in the second image one circle near the image border was missed.

5 Concluding Remarks

The starting point for this work has been the observation that one of the main problems of image processing in log-polar sampled images is that straight lines and distances are not preserved. To overcome this difficulty, we have derived computationally efficient methods to perform the construction of straight lines and the measurement of Euclidean distances in the log-polar plane. These methods were then used as basic operations to implement line and circle detection in log-polar images. Results on synthetic and real images were shown and the accuracy of the detection results were assessed empirically. A direct comparison with detection in conventional images is not possible due to the space-variant resolution, but the obtained results are good enough for many vision tasks.

The log-polar imaging geometry naturally focuses the visual attention and the capability to recover the scene geometry accurately in the image center due to the higher resolution. For many applications in robotics it will therefore be necessary to combine the presented techniques with eye movements, so that the relevant part of the scene is scanned with the image center and can be successively reconstructed with high accuracy. The presented scheme may be useful in such a case, because it extracts features in the periphery of the field of view, too, albeit with comparatively lower accuracy. These features are a good indicator for potentially interesting areas and can be used to guide the eye movements and refine the features.

The ability to construct straight lines in log-polar images has another natural application. Given the camera motion or the relative position and orientation of two log-polar cameras, corresponding epipolar lines can be constructed with the method, which will potentially simplify the stereo matching problem, a notoriously cumbersome task in log-polar images. This is ongoing research.

An investigation, how the results of log-polar image processing are affected by image measurement noise and errors, is still missing. Both a sound theory of the error propagation from log-polar images to the (usually Cartesian) coordinate system of the observed scene and an experimental evaluation of the effects of noise would be valuable contributions towards the use of log-polar sensors in practical applications. In the same line of research it would also be interesting to design an error functional, which allows to optimize the detected features in the log-polar domain.

Acknowledgments

I would like to thank Horst Bischof for valuable comments and the Austrian Research Council (ÖFG) for travel funds. The 'corridor' sequence is provided by the University of Oxford Visual Geometry Group on their web-page.

References

1. Weiman, C., Chaikin, G.: Logarithmic spiral grids for image processing and display. Computer Graphics and Image Processing (1979) 197–226
2. Sandini, G., Tagliasco, V.: An anthropomorphic retina-like structure for scene analysis. Computer Vision, Graphics and Image Processing **14** (1980) 365–372
3. Tistarelli, M., Sandini, J.: On the advantages of polar and log-polar mapping for direct estimation of the time-to-impact from optical flow. IEEE Transactions on Pattern Analysis and Machine Intelligence **15** (1993) 401–411
4. Sandini, G., Alaerts, A., Dierckx, B., Ferrari, F., Hermans, L., Mannucci, A., Parmentier, B., Questa, P., Meynants, G., Scheffer, D.: The project SVAVISCA: A space variant color CMOS sensor. In: Proc. SPIE International Conference on Advanced Focal Plane Arrays and Electronic Cameras, Zürich, Switzerland. (1998)
5. Tunley, H., Young, D.: First order optical flow from log-polar sampled images. In: Proc. 3rd European Conference on Computer Vision, Stockolm, Sweden. (1994) 132–137
6. Salgian, G., Ballard, D.: Visual routines for vehicle control. In Kriegmann, D., Hager, G., Morse, S., eds.: The Confluence of Vision and Control. Springer Verlag (1998)
7. Capurro, C., Panerai, F., Sandini, G.: Dynamic vergence using log-polar images. International Journal of Computer Vision **24** (1997) 79–94
8. Grosso, E., Tistarelli, M.: Log-polar stereo for anthropomorphic robots. In: Proc. 6th European Conference on Computer Vision,Dublin, Ireland. (2000) 299–313
9. Bernardino, A., Santos-Victor, J.: A binocular stereo algorithm for log-polar foveated systems. In: Proc. 2nd workshop on Biologically Motivated Computer Vision, Tübingen, Germany. (2002)
10. Young, D.: Straight lines and circles in the log-polar image. In: Proc. 11th British Machine Vision Conference, Bristol, UK. (2000) 426–435
11. Daniel, M., Whitteridge, D.: The representation of the visual field on the cerebral cortex in monkeys. Journal of Physiology (1961) 203–221
12. Schwartz, E.: A quantitative model of the functional architecture of human striate cortex with application to visual illusion and cortical texture analysis. Biological Cybernetics (1980) 63–76
13. Canny, J.: A computational approach to edge detection. IEEE Transactions on Pattern Analysis and Machine Intelligence **8** (1986) 679–698
14. Rothwell, C.A., Mundy, J.L., Hoffman, W., Nguyen, V.D.: Driving vision by topology. In: Proc. International Symposium on Computer Vision. (1995) 395–400