

Designing the Lattice for Log-Polar Images*

V. Javier Traver and Filiberto Pla

Dep. de Llenguatges i Sistemes Informàtics
Universitat Jaume I, E12071-Castelló, Spain
{vtraver, pla}@uji.es

Abstract. Log-polar images have been being used for pattern recognition and active vision tasks for some years. These images are obtained either from true retina-like sensors or from conventional cartesian images by software conversion. From the hardware perspective, the design of such log-polar retinæ faces its own technological limitations. In the case of software remappers, however, their very flexibility has led to many researchers to use them with little or no justification of the choice of the particular log-polar layout. In this paper, a set of design criteria are proposed, and an approach to choose the parameters involved in the log-polar transform is described. This kind of design not only could be used in simulation software, but also could act as design guidelines for artificial hardware-built retinæ.

Keywords: Log-polar transform, receptive fields, design criteria.

1 Introduction

Motivation. After its biological foundations [12], the log-polar image representation has been adopted in fields of computer vision such as pattern recognition [16] and active vision [4]. Three basic techniques exist for obtaining log-polar images:

- **Hardware retinæ**, which involve the design and manufacture of very specialized sensors which directly yield log-polar frames [6, 17, 9].
- **Software remappers**, which implement the log-polar transform by taking as input conventional cartesian images [8, 3, 13].
- **Virtual sensors**, which simulates the log-polar mapping via special-purpose hardware, also with cartesian images as input [7, 5].

The main challenge faced by the first choice (true log-polar sensors), relates to technological obstacles during their design and fabrication, which, in part, have been overcome over time [11]. In contrast, software-based simulations of the log-polar transform have an amazing flexibility, allowing an easy implementation of different log-polar models each with a variety of designs.

Traditionally, however, scarce attention has been paid to the selection or justification of the proper values for the parameters of the log-polar transform.

* Research partly funded by *Conselleria d'Educació, Cultura i Ciència, Generalitat Valenciana*, under project CTIDIB/2002/333.

This fact seems particularly apparent in the case of software conversion, probably due to the very nature of software.

Importance. Nevertheless, criteria for the design of the log-polar layout would be of great help for the practitioners of computer vision interested in *using* log-polar imagery. Furthermore, the choice of the log-polar geometry has an impact on the performance of the algorithms used in certain tasks. Finally, this kind of design guidelines might also benefit the conception of new silicon retinae.

Related Work. Among the scarce work addressing the topic of log-polar design, we can mention [2], where it is studied how log-polar mapping parameters affect the performance of a vergence control algorithm. In [14], the log-polar sensor design is driven by the relation between the geometry parameters and 3D sensing precision requirements. Several alternatives for the more specific problem of fovea design are proposed in [15]. Quantitative measures of the quality of log-polar sensors are given in [10, 11]. Although not directly related to the geometric design, these measures are useful for comparison between different sensors. The importance of a good choice for the transform parameters is stressed and considered in [1].

Our Work and Structure of the Paper. With respect to these works, we propose a set of general design criteria, and a means to find the transform parameters meeting these criteria. The rest of the paper is organized as follows. Section 2 describes the log-polar model and its parameters. This is the model on which design criteria are discussed in Section 3. Based on these design considerations, parameters of the transform can be selected as described in Section 4. Finally, concluding remarks are given in Section 5.

2 Log-Polar Mapping

2.1 Definition and Basic Parameters

Among the different log-polar image representations proposed in the literature, we choose the *central blind-spot model* because of its interesting properties [13] (e.g., retinal rotations and scalings map both to simple shifts in the cortical plane). Under this model, the log-polar coordinates are defined as:

$$(\xi, \eta) \triangleq \left(\log_a \left(\frac{\rho}{\rho_0} \right), \theta \right), \quad (1)$$

with (ρ, θ) being the polar coordinates defined from the cartesian coordinates (x, y) as usual, i.e., $(\rho, \theta) \triangleq \left(\sqrt{x^2 + y^2}, \arctan \frac{y}{x} \right)$. Because of the discretization, the continuous coordinates (ξ, η) become the discrete ones $(u, v) = (\lfloor \xi \rfloor, \lfloor q \cdot \theta \rfloor)$, $0 \leq u < R$, $0 \leq v < S$, with R and S being the number of rings and sectors of the discrete log-polar image, and $q = \frac{S}{2\pi}$ sectors/radian. The notation $\lfloor z \rfloor$ denotes the common floor operation, i.e., the largest integral value not greater than z . Having chosen R , ρ_0 (the radius of the innermost ring), and ρ_{\max} (the radius of the visual field), the transformation parameter a is computed as $a = \exp(\ln(\frac{\rho_{\max}}{\rho_0})/R)$. If the original cartesian image is sized $M \times N$, ρ_{\max} can be

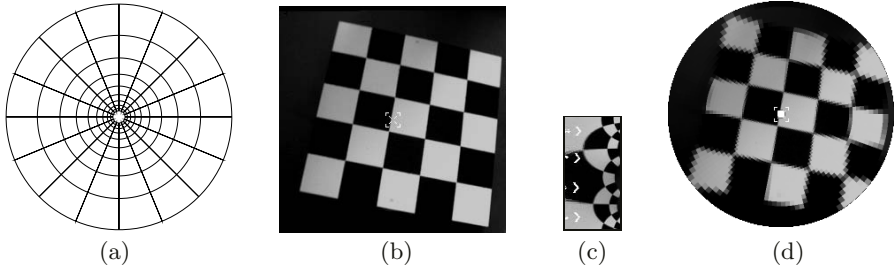


Fig. 1. Log-polar mapping: (a) grid layout example (10×16), (b) original cartesian image (256×256), (c) cortical image (64×128), (d) retinal image (256×256) obtained by the inverse mapping from (c)

defined as $\rho_{\max} = \frac{1}{2} \min(M, N)$, and the log-polar transform is centered at the foveation point $(x_c, y_c) = (\frac{M}{2}, \frac{N}{2})$.

Illustrative Example. An example of a log-polar transformation is shown in Fig. 1, from which several observations can be made. First of all, it can be appreciated the much smaller size of the cortical image (Fig. 1(c)) compared to the original uniformly-sampled image (Fig. 1(b)), which illustrates the data reduction property. Second, the small arrows radially disposed in the cartesian image become magnified and parallel one to each other (see Fig. 1(c)), which demonstrates how (i) visual acuity is higher in the fovea area, and (ii) rotations become translations along the angular axis. Third, note in the retinal visualization of the cortical image (Fig. 1(d)) how edges near the image center are much sharper than edges at the periphery, because of the space-variant resolution.

2.2 Derived Properties

From the basic parameters involved in the log-polar transform, we define other parameters which are a quantification of some properties of the log-polar layout. These measures will later be used in Section 3 for formalizing the design criteria.

Log-Polar Image Size. It is simply the total number of pixels, i.e., $N = R \cdot S$.

Aspect Ratio of Receptive Fields. The aspect ratio of a geometric entity is the ratio between its width and its height. Given that a receptive field (RF) is not rectangular, its width is not well-defined. As an approximation, we can consider its outer or its inner boundary as its width (or even a function of both of them). Here, the length of the inner arc will be chosen as the RF's width (see Fig. 2).

With these considerations, for any RF at the same eccentricity u , we have that its width will be given by (arc equals angle per radius):

$$w(u) = \frac{2\pi}{S} \cdot \rho_{u-1} = \frac{2\pi}{S} \cdot \rho_0 \cdot a^{u-1}, \quad (2)$$

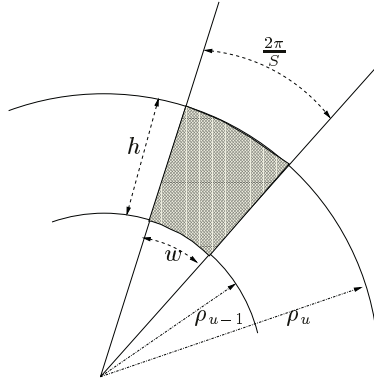


Fig. 2. Geometric elements involved in the computation of the area and the aspect ratio of receptive fields

and its height by

$$h(u) = \rho_u - \rho_{u-1} = \rho_0 \cdot a^u - \rho_0 \cdot a^{u-1} = \rho_0 \cdot a^{u-1}(a - 1). \quad (3)$$

Therefore, the aspect ratio $\gamma(u) = \frac{w(u)}{h(u)}$ is:

$$\gamma = \frac{w}{h} = \frac{\frac{2\pi}{S} \cdot \rho_0 \cdot a^{u-1}}{\rho_0 \cdot a^{u-1}(a - 1)} = \frac{\frac{2\pi}{S}}{a - 1} = \frac{2\pi}{S(a - 1)}. \quad (4)$$

By observing Eq. 4, it can be noticed that the aspect ratio is *not* a space-variant quantity: all RFs in a log-polar grid have the same aspect ratio. This interesting result, however, does not hold in other log-polar models (e.g., in Jurie’s model [8]).

Area of Receptive Fields. Because of the space-variant nature of the log-polar geometry, RFs at different eccentricities cover a different surface. The area of a RF can be found by computing the area of a circular annulus and then dividing the result by the number of sectors. The area of a circular annulus at eccentricity u is $\pi \cdot \rho_{u+1}^2 - \pi \cdot \rho_u^2 = \pi(a^2 \cdot \rho_u^2 - \rho_u^2) = \pi \cdot \rho_u^2(a^2 - 1)$. Then, the area σ of a single RF will be a S -th part of this, i.e.:

$$\sigma(u) = \frac{\pi \cdot \rho_u^2(a^2 - 1)}{S}. \quad (5)$$

Oversampling. In a software-based implementation of the log-polar mapping, RFs near the center (the fixation point) become much smaller than the cartesian pixels (Fig. 3(b)). As a result, information contained in cortical images at the center, where most pixels are, become highly redundant. This situation is known as “oversampling”, because cartesian images are oversampled, i.e., sampled at

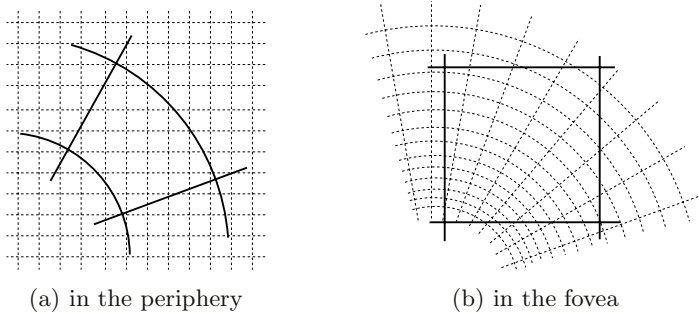


Fig. 3. Relative size between RFs and cartesian pixels: in (a) RFs (in dark solid lines) are larger than cartesian pixels (in dotted lines); in (b) cartesian pixels (in dark solid lines) are larger than RFs (in dotted lines)

a frequency higher than its maximum frequency. Undersampling also occurs at periphery, where cartesian pixels and RFs happen to be in a many-to-one relationship (Fig. 3(a)). However, this undersampling is not only desired, but it is the very essence of the selective data reduction of discrete log-polar images.

The maximum oversampling occurs at the innermost ring ($u = 0$), and this is the one which it is interesting to be quantified. Note that the area of a RF at ring u , $\sigma(u)$, is expressed as $\frac{\text{No. cartesian pixels}}{1 \text{ receptive field}}$. Because near the fovea (i.e., for small u), this ratio tend to become smaller than the unity, we choose the inverse of the area at $u = 0$ as the (maximum) oversampling, so that, its units become $\frac{\text{No. receptive fields}}{1 \text{ cartesian pixel}}$, which is closer in meaning to the oversampling effect. Thus, the oversampling is quantified as

$$o = \sigma(0)^{-1}, \quad (6)$$

i.e., the number of RFs covering a single underlying cartesian pixel.

3 Design Criteria

Limiting Computational Complexity. In computer vision applications, running time and memory space requirements are proportional to the number of pixels of the images to be processed. Therefore, a simple way of bounding computational resources can be modeled as $N < N_{\max}$, i.e., imposing that the size of the log-polar image keeps under a certain value (N_{\max}), according to the computational power available.

Having Unit Aspect Ratio RFs. In theory, it is possible to choose any combination for R and S . Even with the above constraint, there are many possibilities for these parameters. However, not all of these combinations result in “good” log-polar grids. As an example, see Fig. 4, where different layouts are shown,

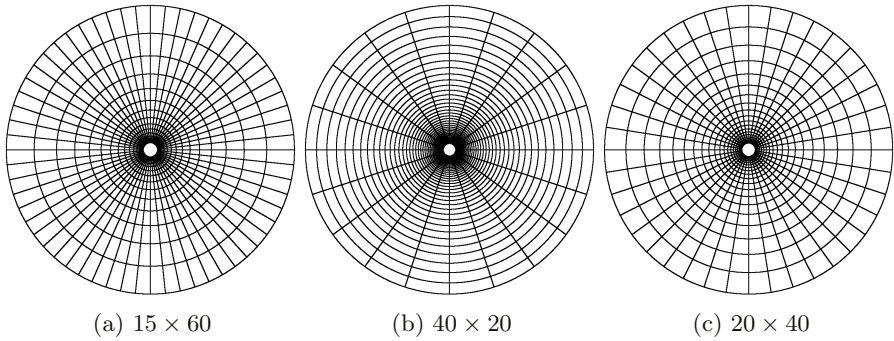


Fig. 4. Log-polar grids with different aspect ratios in the RFs' geometry: (a) $\gamma < 1$, (b) $\gamma > 1$, (c) $\gamma \approx 1$

which illustrate what the problem can be. In one case (Fig. 4(a)), pixels are too much “elongated” (when $R \ll S$); in the other (Fig. 4(b)), pixels are too much “flattened” (when $R \gg S$). The problem in both cases is the same: the log-polar pixels (actually, their associated receptive fields) have a “wrong” aspect ratio, either too small or too big. We believe this is an undesirable feature when applying some operations on the image, because neighbor RFs would be at different distances along the radial and angular directions. From a different point of view, we might be interested in having comparable resolutions in both the radial and angular axes.

Therefore, preserving the aspect ratio close to 1, i.e., having RFs approximately squared (Fig. 4(c)), allows local image processing operators be applied correctly, and log-polar images mapped back to the cartesian domain are perceptually better. Then, by forcing $\gamma = \frac{w}{h} = 1$, we have $\gamma = \frac{2\pi}{S(a-1)} = 1$. To fulfill this constraint, we can choose S as a function of R (or vice versa). As a is already a function of R , we can write:

$$S = \frac{2\pi}{a-1} \triangleq S_\gamma. \quad (7)$$

Minimizing Oversampling. In some applications, oversampling is not desirable or, at least, should not be too big. Ideally, null oversampling would be achieved with $o = \sigma(0)^{-1} = \left(\frac{\pi \cdot \rho_0^2 (a^2 - 1)}{S}\right)^{-1} = 1$. Then,

$$S = \pi \rho_0^2 (a^2 - 1) \triangleq S_o. \quad (8)$$

Preserving Small Objects Observable. The log-polar model considered in this paper is characterized by having a central blind spot (as can be seen in Fig. 1(a) and Fig. 4). This implies that objects centered in the visual field will only be detectable if they are bigger than the sensor's blind area. Therefore, if objects with radius r_{\min} are to be observed (at least partially), we should have $\rho_0 < r_{\min}$.

Having a Wide Field of View (FOV). Just one of the appealing properties of log-polar images is that they offer a good compromise between three important requirements: resolution, image size and FOV. The FOV's width can be modeled as ρ_{\max} .

4 Finding the Mapping Parameters

Free parameters. There are a total of six parameters $\Psi = (R, S, \rho_0, \rho_{\max}, a, q)$ involved in the log-polar model. In principle, these are the parameters whose values should be found. However, R , ρ_0 , ρ_{\max} , and a are related, so that each of them can be found from the three others. Second, ρ_{\max} will be considered a fixed parameter because: (i) the log-polar image is computed from cartesian images, so the size of the latter gives the value for ρ_{\max} ; (ii) this parameter only affects the scale of the sensory layout. Third, q is a function of S . Finally, with the design criteria considered above, S can be found from R . Therefore, only two parameters remain free: R and ρ_0 .

Trading criteria. As usual with any design process, there are conflicting criteria: to observe small targets, ρ_0 should be small; to have small oversampling, ρ_0 should be large. Therefore, a trade-off solution is required. We propose the use of a user-selectable parameter, λ , which weighs the relative importance given to these two criteria. Notice that these criteria were expressed as constraints on the values of S (Eqs. 7 and 8). On the other hand, it can be shown that $S_o < S_\gamma$. Then, we suggest to use the following combined constraint:

$$S = S_o + \lambda(S_\gamma - S_o), \quad \lambda > 0.$$

Therefore, the lower λ is, the more importance is given to null oversampling. The closer λ is to 1, the more importance is paid to unit aspect ratio RFs. Regarding the aspect ratios, $\gamma > 1$, for $\lambda < 1$, and $\gamma < 1$ for $\lambda > 1$.

Algorithm. Going a step further, a mere user of the log-polar transformation should be concerned as little as possible about particularities and details of the log-polar model being used. Therefore, the user requirements should be expressed in terms of higher-level design criteria. To that end, we propose a simple procedure (Algorithm 1) in which the input from the user is:

- N_{\max} , the approximate number of pixels for the resulting log-polar image;
- α , the allowable error between N_{\max} and the total number of pixels N found;
- r_{\min} , the size of the smallest object that should be visible; and
- λ , the trade-off value weighting small oversampling versus close-to-one aspect ratio RFs.

This algorithm proceeds iteratively. Initially, an estimate is set for R from the required N_{\max} (e.g., assuming $R = S$). At each iteration, a new set of mapping parameters, Ψ , is computed, and R is updated from the newly found N , using

```

ComputeMappingParameters( $R, \rho_0, \lambda$ ) :  $\Psi(R, S, \rho_0, \rho_{\max}, a, q)$ 
   $a \leftarrow \exp(\ln(\frac{\rho_{\max}}{\rho_0})/R)$  //  $\rho_{\max}$  considered given a priori
   $S_o \leftarrow \pi \rho_0^2 (a^2 - 1)$  // Null oversampling
   $S_\gamma \leftarrow 2\pi/(a - 1)$  // Unit aspect ratio RFs
   $S \leftarrow S_o + \lambda(S_\gamma - S_o)$  // Compromise solution
   $q \leftarrow \frac{S}{2\pi}$ 
  return  $\Psi(R, S, \rho_0, \rho_{\max}, a, q)$ 

FindGoodDesign( $N_{\max}, \alpha, r_{\min}, \lambda$ ) :  $\Psi(R, S, \rho_0, \rho_{\max}, a, q)$ 
   $\rho_0 \leftarrow r_{\min}/2$  // Any convenient function of  $r_{\min}$ 
   $R \leftarrow \sqrt{N_{\max}}$  // Any reasonable initial guess
  repeat
     $\Psi \leftarrow \text{ComputeMappingParameters}(R, \rho_0, \lambda)$ 
     $\beta \leftarrow \Psi_R \cdot \Psi_S / N_{\max}$ 
     $R \leftarrow R/\beta$  // Rectify estimate
  until  $|\beta - 1| < \alpha$ 
  return  $\Psi(R, S, \rho_0, \rho_{\max}, a, q)$ 
    
```

Algorithm 1: Finding the parameters of a log-polar layout from design criteria formalized quantitatively

the amount of deviation, β , as a corrective factor. The process is repeated until N and N_{\max} are close enough (according to how demanding the user has been by specifying α). Notice that the value of a particular parameter of the 6-tuple Ψ is denoted using the name of that parameter as a subindex (e.g., Ψ_R is the value of R in Ψ).

Examples. Table 1 shows four examples of input parameters and results obtained by using the algorithm described above. The resulting real values for R and S have been rounded to the nearest integers, and o and γ have been approximated to 2 decimal places.

Table 1. Design examples: input requirements and resulting parameters and measures

EXAMPLE	INPUT CRITERIA				RESULTS						
	N_{\max}	r_{\min}	α	λ	R	S	N	ρ_0	o	γ	# iters.
1	2000	20	0.1	0.9	28	65	1820	10	1.03	1.00	13
2	2000	10	0.1	0.9	32	56	1792	5	3.20	1.08	127
3	2000	10	0.1	0.1	93	23	2139	5	4.04	7.77	3
4	4000	5	0.1	0.7	63	69	4347	2.5	26.62	1.41	1

In the first example, because r_{\min} is relatively high, it was possible to have both small oversampling and unit aspect ratio RFs. The total number of pixels ($N = 1820$) differs in less than 10% (as specified with $\alpha = 0.1$) from $N_{\max} = 2000$. In the second example, r_{\min} is smaller, and it can be appreciated that the algorithm takes longer (127 *vs.* 13 iterations) to find a good combination of mapping parameters. It is interesting to appreciate that, because the criterion of

having unit aspect ratio is stressed more than having low oversampling (because $\lambda = 0.9$), we get a good aspect ratio ($\gamma = 1.08$), but oversampling has increased ($o = 3.2$).

In the third example, less attention is paid to having unit aspect ratio RFs ($\lambda = 0.1$), this resulting in a very high, unreasonable aspect ratio. In the fourth example, an intermediate trade-off value $\lambda = 0.7$ is provided. This example would be of a design meant for visualization purposes: higher number of pixels ($N_{\max} = 4000$), and small observable targets ($r_{\min} = 5$). In this case, oversampling is not an issue, because its effect is not visually perceivable, it only affects redundancy in data in the log-polar image. Notice the very high oversampling we incur in this case ($o = 26$).

Comments. The idea of an algorithm processing high-level design specifications and yielding low-level mapping parameters is very attractive. In this sense, Algorithm 1 represents an effort along the line of automating the design process. However, because the procedure was basically driven to achieve a given total number of pixels, other criteria could not easily be met at the same time, or compromise solutions are not dealt conveniently. In practical terms, for certain input requirements, this algorithm might not find a solution (and waste iterations in the attempt), or give up with improvable solutions in just one or a few iterations (like example 4 in Table 1).

5 Conclusions

Little effort has been paid in the past to a proper selection of the parameters of the log-polar mapping, in particular when the transform is implemented in software. After a brief description of a log-polar model, this paper examines possible design criteria that should guide the choice of the values of the mapping parameters. A mathematical expression has been derived for each criterion. Then, it is discussed how these design constraints could be met. Because the different criteria lead to contradictory goals, only trade-off solutions are possible. Although the design process can be completely trial-&-error-based, an algorithm has been proposed to help designers and end users find reasonably adequate solutions. Interestingly, the input are specified as high-level design requirements. Further work could be directed to develop some algorithm which considers and explores the design space more effectively.

References

1. R. Alan Peters II, M. Bishay, and T. Rogers. On the computation of the log-polar transform. Technical report, School of Engineering, Vanderbilt University, Mar. 1996. <http://www.vuse.vanderbilt.edu/~rap2/papers/oncomplp.pdf>.
2. A. Bernardino and J. Santos-Victor. Sensor geometry for dynamic vergence: Characterization and performance analysis. In *Workshop on Performance Characteristics of Vision Algorithms, ECCV*, 1996. (Also as TR 01/96 at VisLab, Lisbon, Portugal).

3. A. Bernardino and J. Santos-Victor. Visual behaviors for binocular tracking. *Robotics and Autonomous Systems*, 25:137–146, 1998.
4. M. Bolduc and M. D. Levine. A review of biologically motivated space-variant data reduction models for robotic vision. *Computer Vision and Image Understanding (CVIU)*, 69(2):170–184, Feb. 1998.
5. J. R. del Solar, C. Nowack, and B. Schneider. VIPOL: A virtual polar-logarithmic sensor. In *Scandinavian Conf. on Image Analysis (SCIA)*, pages 739–744, Finland, 1997.
6. J. V. der Spiegel, G. Kreider, C. Claeys, I. Debusschere, G. Sandini, P. Dario, F. Fantini, P. Belluti, and G. Soncini. A foveated retina-like sensor using CCD technology. In C. Mead and M. Ismail, editors, *Analog VLSI and Neural Network Implementations*, Boston, 1989. DeKluwer Publ.
7. T. E. Fisher and R. D. Juday. A programmable video image remapper. In *SPIE Conf. on Pattern Recognition and Signal Processing*, volume 938 (Digital and Optical Shape Representation and Pattern Recognition), pages 122–128, 1988.
8. F. Jurie. A new log-polar mapping for space variant imaging. Application to face detection and tracking. *Pattern Recognition*, 32:865–875, 1999.
9. F. Pardo, J. A. Boluda, J. J. Pérez, B. Dierickx, and D. Scheffer. Design issues on CMOS space-variant image sensors. In *SPIE Conf. on Advanced Focal Plane Arrays and Electronic Cameras (AFPAEC)*, Berlin, Germany, Oct. 1996.
10. A. S. Rojer and E. L. Schwartz. Design considerations for a space-variant visual sensor with complex-logarithmic geometry. In *Intl. Conf. on Pattern Recognition (ICPR)*, pages 278–285, 1990.
11. G. Sandini, P. Questa, D. Scheffer, B. Dierickx, and A. Mannucci. A retina-like CMOS sensor and its applications. In *Proc. 1st. IEEE SAM Workshop*, Cambridge, USA, Mar. 2000.
12. E. L. Schwartz. Spatial mapping in the primate sensory projection: Analytic structure and relevance to perception. *Biological Cybernetics*, 25:181–194, 1977.
13. V. J. Traver. *Motion Estimation Algorithms in Log-polar Images and Application to Monocular Active Tracking*. PhD thesis, Dep. Llenguatges i Sistemes Informàtics, Universitat Jaume I, Castellón (Spain), Sept. 2002.
14. C. F. R. Weiman. Exponential sensor array geometry and simulation. In *Preprint from the Proc. of SPIE*, Orlando, Florida, Apr. 1988. Vol. 938 (Digital and Optical Shape Representation and Pattern Recognition).
15. C. F. R. Weiman. Log-polar binocular vision system. NASA Phase II SBIR Final Report, Dec. 1994.
16. J. C. Wilson and R. M. Hodgson. Log-polar mapping applied to pattern representation and recognition. *Computer Vision and Image Processing*, pages 245–277, 1992.
17. R. Wodnicki, G. W. Roberts, and M. D. Levine. A foveated image sensor in standard CMOS technology. In *Custom Integrated Circuits Conf.*, Santa Clara, May 1995.