Automatic Camera Exposure Control

Navid Nourani-Vatani[1] **and Jonathan Roberts**[2]
[1]Tasmanian ICT Centre
[2]Autonomous Systems Laboratory
CSIRO ICT Centre, Australia
{Navid.Nourani, Jonathan.Roberts}@csiro.au

Abstract

It is commonplace to use digital video cameras in robotic applications. These cameras have built-in exposure control but they do not have any knowledge of the environment, the lens being used, the important areas of the image and do not always produce optimal image exposure. Therefore, it is desirable and often necessary to control the exposure off the camera. In this paper we present a scheme for exposure control which enables the user application to determine the area of interest. The proposed scheme introduces an intermediate transparent layer between the camera and the user application which combines the information from these for optimal exposure production. We present results from indoor and outdoor scenarios using directional and fish-eye lenses showing the performance and advantages of this framework.

1 Introduction

Camera exposure control is one of the most essential part of still-photography and filming to ensure all the details being visible. Consequently, it has always been a research priority for camera manufacturers to develop better and more accurate exposure control methods. A simple search for "camera exposure patent" in Google Scholar returns 55,000 results. The majority of the technologies developed are hardware based. Initially external hardware was used to perform the scene light measurement. Later through development of TTL (Through-The-Lens) technologies this unit became part of the camera itself, but still hardware based [Nikon, 2007]. The built-in exposure control in the camera tries to change the exposure so it fits a middle grey tone; a so called 18% grey[1]. The most common method to measure the light in the image is to perform a centre-weighted average metering [Reichmann, 2007]. This method averages the exposure in the entire image while assigning more weight to the central $60-80\%$ of the image. This works well in most situations, but in difficult lighting situations, especially outdoor usage, and with different lenses this metering method will fail. Other metering modes found in higher-end cameras include spot-metering, evaluative metering, and matrix metering [Canon, 2007; Nikon, 2007]. The primary techniques mainly differ in the area they use for their metering, e.g. spot metering uses 4% of the centre of the image, while evaluative metering averages over the entire image [McHugh, 2007]. The matrix metering technologies are more advanced and use a type of honeycomb configuration to pick out objects in the image and thereby perform different exposure over the image. In most recent digital cameras some Artificial Intelligence has been introduced in the form of face detection. The face detection feature is used by the camera to ensure correct exposure - and focus - of the area of importance; the detected faces.

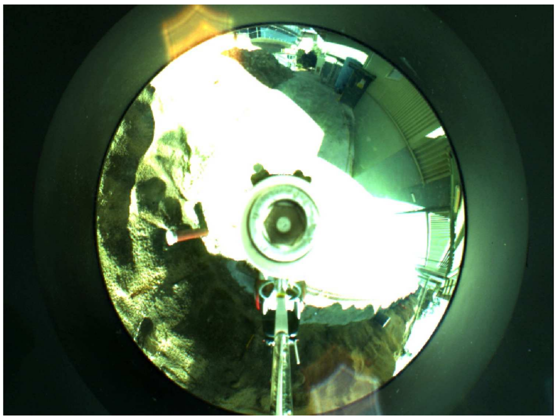


Figure 1: Auto exposed frames using omni-directional lens. The centre of the image is overexposed as the controller is trying to compensate for the dark areas surrounding the mirror.

[1]18% grey can be represented in the sRGB space approximately with the values R,G,B = 118,118,118

Essentially it is the photographer, accessing these metering modes and using the knowledge of lighting condition, subject, and area of interest who captures a correctly exposed image. In robotics it is now becoming commonplace to use omni-directional and fish-eye lenses. These lenses create round images in a rectangular frame resulting in dark circumference. Allowing the camera to control the exposure will result in overexposure of the real image due to this large dark area (Figure 1). Issues are also created due to the mirror not being centered exactly in the image, and there are often parts of the robot itself, such as the mirror mounting bracket, always visible in the frame - these need to be removed. Hence, the area of interest is often known, can be predicted, or can be calculated. In other words, there can be regions that we want to disregard and make sure does not influence the exposure of the image. Another issue is that of bright sun and image flaring which disturbes the exposure. This issue is discussed later in the paper. The areas of interest also change with time, and hence the metering mode should change as well to continue capturing correctly exposed images. In this paper we propose a scheme which allows for this flexibility, while facilitating the work of the user application.

The reminder of this paper is structured as follows. Section 2 describes our approach to exposure control and explains how this application fits within the software framework. Section 3 shows the performance of our approach and results achieved. Finally, Section 4 lists some conclusions and proposes future work.

2 Exposure Control

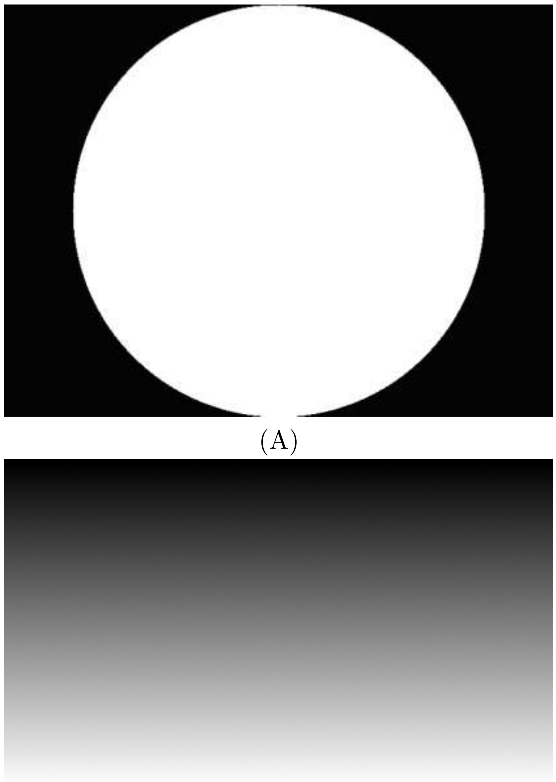
2.1 Masks

Standard metering modes are very rigid in their determination of area of interest. Photographers overcome this limitation by incorporating their knowledge of the scene and lighting, by changing their view of capture, and by choosing the area of metering. But in real-time image processing on a mobile platform these measures cannot be used. Even when the area of interest is known there is no means by which this knowledge can be passed to the unit controlling the exposure. It is therefore necessary to have a layer between the camera and the user application to manage this. One approach is to perform exposure bracketing. In this scheme several images with different exposures are taken. The correctly exposed areas of these images are then combined to one correctly exposed image. This technique has been successfully implemented for robotics vision by [Nuske *et al.*, 2006]. The downside of this method is the spatial discrepancies between the images and the processing overload.

We introduce the usage of masks to enable the user application to determine the area of interest. A mask is an image with the same size as the camera frame and is often used in image processing to carry information about area of interest, i.e. in OpenCV's copy function, `cvCopy( src, dst, mask )`, a binary mask can be used to tell which pixels should be copied from the source image to the destination image [Intel Corporation, 2001]. Masks do not restrict the area of interest to be in the center of the image and allow the user to select any shape as the area of interest. We recommend using a single layer image with a depth of 8 bits. This gives sufficient flexibility to implement gradient masks. Figure 2 shows some commonly used masks.

2.2 Image Histograms

With the age of digital imaging a simple but very powerful tool has emerged in photography, namely image histogram. An image histogram is a graphical repre-



(A)

(B)

Figure 2: Examples of commonly used masks. Here white pixels represent areas of interest and black pixels represent areas to be disregarded. For gradient masks the pixel gray-scale value is used as a weight. (A) A circle mask is used with fish-eye and omni-directional lenses. (B) A vertical gradient mask can be used to minimize the effect of bright sky.

sentation of the brightness in the image shown as bars corresponding to the count of the luminance value in the image ranging from 0 to 255[2]. This basic tool can be used for image segmentation and thresholding [Horn, 1986]. Image histograms can also indicate the nature of the lighting conditions, the exposure of the image, and whether it is underexposed or overexposed. Figures 3(A) and (B) show a test image and the corresponding histogram. The histogram can be divided into 5 regions as shown in the figure. These regions represent the dynamic response range of the film, or now in our case, the response of the camera sensor. The left regions represent dark colors while the right regions represent light colors. An underexposed image will be leaning to the left while an overexposed image will be leaning to the right in the histogram [Reichmann, 2007]. These situations are shown in Figures 4(A) and (B). From these figures it is obvious how the image details disappear when underexposed or overexposed. Hence, we want as much as the image to appear in the middle region of the histogram. But the histogram is only a visual representation for humans. To use it for automatic exposure control these bars have to be interpreted by another means. One measure to use is the mean sample value [Shirvaikar, 2004]. The mean sample value (MSV) calculated from the histogram determines the balance of the tonal distribution in the image:

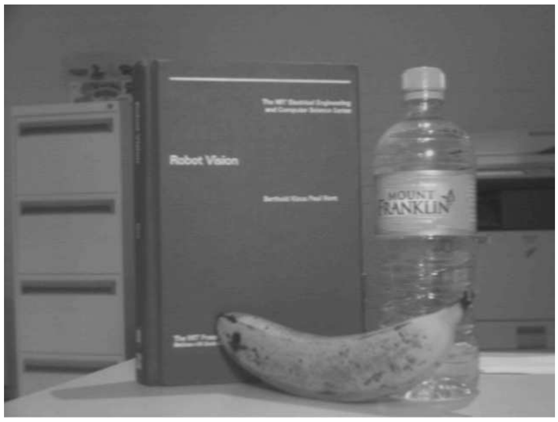
$$\mu = \frac{\sum_{i=0}^{4}(i+1) \cdot x_i}{\sum_{i=0}^{4} x_i} \tag{1}$$

where x_i is the sum of the values in region i. In Figures 3 and 4, the *val* represents the MSV for each histogram. As the name implies, MSV is a mean measure which does not take into account regional overexposures and underexposures in the image. To cope with these situations more analysis of the histogram or the image needs to be carried out. Using MSV, the image is correctly exposed when $\mu \approx 2.5$ as in Figure 3.

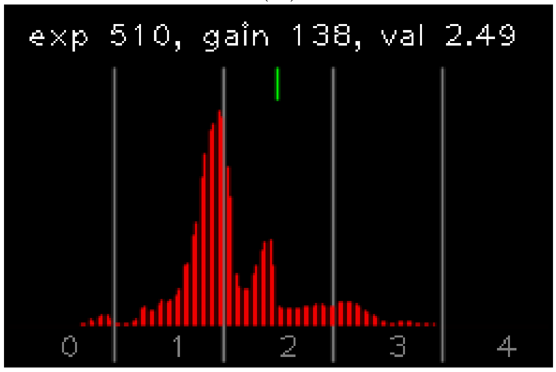
2.3 Technology Framework

Digital webcams today use either USB or firewire connection with their specifically defined protocols. Also defined in the protocols are some parameters to control the behavior of the cameras, e.g. shutter speed, exposure value, frame rate etc. These protocols are very generic and not all the parameters apply to all cameras. It is therefore not possible to develop a generic application to handle the control of all cameras. At the Autonomous Systems Lab (ASL) in Brisbane, one of our cameras of

[2]The interval 0-255 is the most common as most cameras use 8-bit representation. High-end Digital Single Lens Reflection (D-SLR) cameras use often 12-bit and 16-bit representation, which give much bigger ranges $2^{12} = 4096$ and $2^{16} = 65536$



(A)



(B)

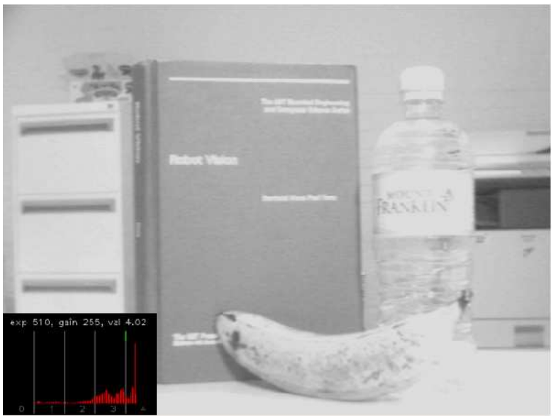
Figure 3: (A) Image of test scenario and (B) the corresponding normalized histogram.

choice is the Unibrain Fire-i camera [Unibrain, 2007]. This camera uses the IEEE 1394 protocol, and allows manual setting of several parameters. The two values of importance are the so called *exposure* and *gain* parameters. As this camera does not have a lens with adjustable aperture the exposure parameter translates to a shutter time. The gain parameter is the sensor sensitivity.

The communication with the camera is dealt with using DDX, which is an in-house developed suite of applications for data and memory sharing and management[Corke *et al.*, 2004] and its video extension DDXVideo[Duff, 2005]. User applications define their region of interest through a mask in DDX. This mask is read by the exposure control application which then sets the appropriate camera parameters for optimal exposure. The exposure control application is working through DDX as well and is totally transparent to the user applications. This facilitates the work of other applications using the camera significantly as they need no knowledge of the camera being used and its control parameters while resulting in better performance. This architecture is shown in Figure 5.

We want to achieve an overall camera Mean Sample

(A)



(B)

Figure 4: Image of test scenario and the corresponding normalized histograms overlaid. (A) Underexposed with $\mu = 1.23$ and (B) Overexposed with $\mu = 4.02$.

Value (MSV) of 2.5. On cameras where the aperture value and shutter time can be set directly, the amount of change to the shutter speed or aperture value can be calculated directly from the histogram as the five regions of the histogram represent five f-stops. Each time the shutter time is doubled/halved the image exposure will decrease/increase with one f-stop. In the same manner, the aperture value can be used to change one f-stop, i.e. decrease by going from f/5.6 to f/8.0 or increase by going from f/5.6 to f/4.0. Most cameras and lenses allow for 1/2 and 1/3 f-stop changes. [Reichmann, 2007]. But no standard is given for the intervals of these parameters in IEEE and therefore mathematical equations cannot be used for determination of the correct exposure and we are forced to use controllers to calculate the optimal values. For the Unibrain Fire-i camera the two camera parameters available, exposure and gain, are dealt with separately. The intervals are 1-511 and 1-127, respectively for the exposure and gain parameters, and for each of these a PI-controller is implemented. PI controllers

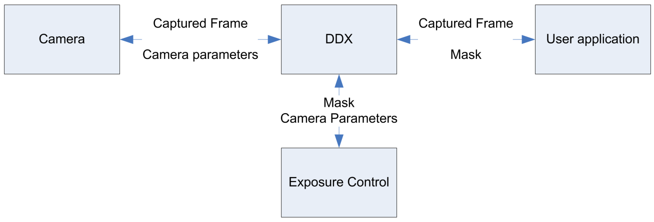


Figure 5: Architecture of applications. The camera exposure control application is transparent to other user applications.

are used over proportional controllers as they result in better control having no stationary error [Jannerup and Soerensen, 2000]. The controller proportional constants are determined empirically as:

$$K_{exposure} = 30 \cdot c + 20 \quad (2)$$

$$K_{gain} = 10 \cdot c + 15 \quad (3)$$

where c is a parameter describing how many percentage of the image is used for calculation of the exposure. It is desirable to keep the gain parameter as low as possible to minimize noise in the image. The controller for this parameter is kept off and the gain parameter value is set to 1 until the exposure parameter reaches maximum. At this point the gain controller is turned on and the controller for the exposure parameter is turned off.

For calculation and normalization of the histogram we use the OpenCV functions `cvCalcHist( &frame, hist, 0, mask )` and `cvNormalizeHist( hist, 1.0 )`. For description and usage of the OpenCV library we refer to their manual [Intel Corporation, 2001].

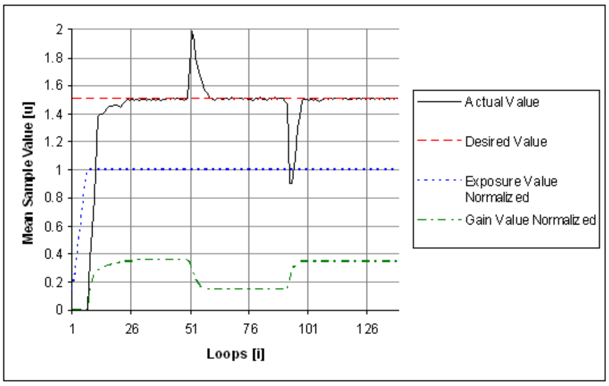
3 Results

We test our system performance by using our indoor test scene. This scene is made up of a red book, a transparent bottle of water and a banana on a white surface. The room is lit by ceiling spot lights and ambient light coming through the windows. The first test scenario has three phases: 1) The exposure control application is started 2) A bright desk lamp pointing at the objects is turned on 3) The light is turned off again. The second scenario also contains three phases: 1) The exposure control application is started 2) A small binary circular mask covering 70% of the image is applied on the fly 3) The mask is removed again. The values for the histogram mean sample value, the camera exposure parameter, and the camera gain parameter value for these tests are shown in Figures 6(A) and (B).

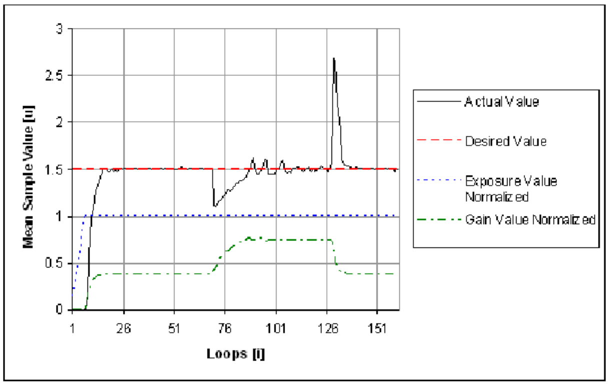
The performance of the control algorithms can be measured by looking at the time it takes for the controller to achieve the correct exposure. The rise time of the controller is normally measured as the time it takes to go from 10% – 90% of the desired value [Jannerup and Soerensen, 2000]. From these examples we notice that approximately 5 cycles are necessary to stabilize the exposure. In test 1, the light turned on results in the peak in the mean sample value. As the exposure parameter is already saturated, this overexposure is compensated for by decreasing the gain parameter. The underexposure is visible from the figure as the light source is turned off again. The gain parameter is increased to re-stabilize the MSV. In test 2, the mask applied results in the surrounding areas of the image not being taken into consideration and therefore a drop in brightness. The gain controller compensates for this underexposure by increasing the gain parameter. As the mask is removed again, the gain parameter is decreased to compensate for the extra areas now again visible. The small overshoot and oscillations in this test are caused by large controller constants and more aggressive controllers, as the control parameter c from Equation 2 is kept at 1.0.

Now consider the situation shown in Figure 7. A fish-eye lens is mounted on the camera looking at the test site outside our office. The sun is not visible but the sky is bright. In Figure 7(A), the camera's exposure control is used while in Figure 7(B) our exposure control is used with the circle mask applied to mask away the area outside the lens. The difference is clear. The camera's onboard exposure controller is trying to compensate for the dark area, which results in the sky and, more importantly, the features on the road, wall and the windows on the right hand side are washed out. Our application knows it must not take the area outside the lens into consideration and therefore produces a much better exposure maintaining all the important features of the image.

The issue with sun flare is more severe when using fish-eye and omni-directional lenses compared to normal directional lenses as the sky is more visible. Having custom exposure control is therefore vital in these situations. Figure 8 shows an example image taken with a fish-eye lens on a sunny day outside our lab. The sun flare is so strong that it produces a vertical line going through the entire image. The camera exposure control cannot deal with these cases but our framework allows detection and masking of the sun.



(A)



(B)

Figure 6: Performance of the controllers. (A) A desk lamp is turned on and then off. (B) A small circular mask is added through DDX and removed again. Note in both figures, that when the controller is initiated the *exposure* parameter is increased very rapidly to its saturation point where it is kept as the ambient light is low. Only after this point the *gain* parameter is controlled.

4 Conclusion

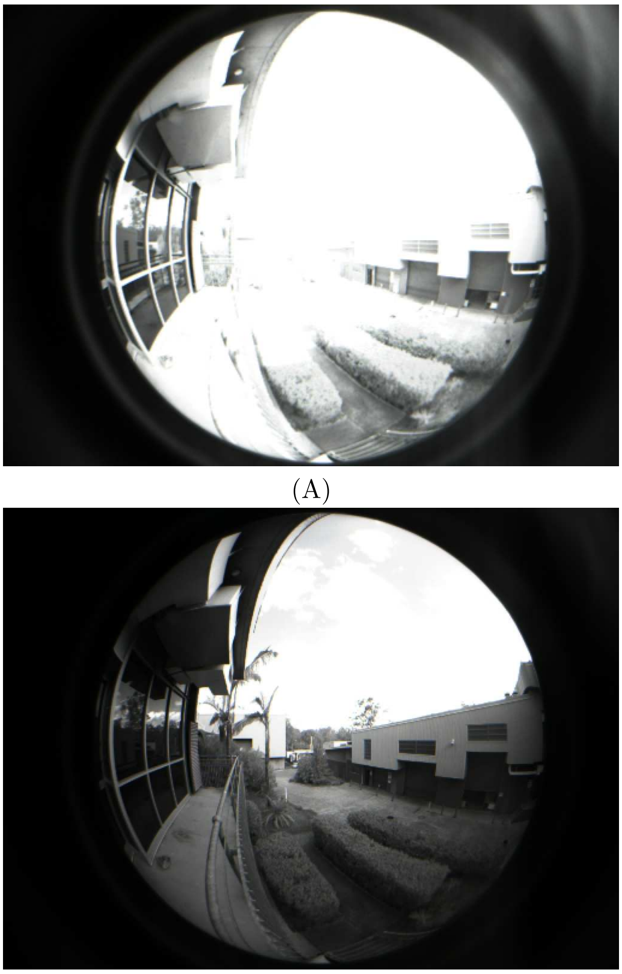
In this paper we have presented a scheme for dealing with the exposure control of digital cameras. We introduced the usage of masks to allow user applications with knowledge of the lens, the environment, and the area of interest to decide which part of the image should be exposed against. The masks can be applied and changed on the run in real-time using our dynamic memory sharing architecture. The control of camera parameters is carried out using separate PI controllers for each parameter. The choice of controller constants is a performance trade-off where we try to avoid overshoots.

Currently, there are no standard for the camera parameter intervals. We are currently changing the DDX video extension to provide normalized camera parameter interfaces - [0:1]. This generalizes and simplifies the control, and adds another level of abstraction.

In the future we need to implement usage of gradient masks to gain more flexibility. Implementation of a simple low-level detection and handling of cut-off areas is also desirable.

Acknowledgments

This work was funded by the CSIRO ICT Centre Autonomous Systems Laboratory, The Tasmanian ICT Centre and the CSIRO Light Metals Flagship. The Tas-



(A)

(B)

Figure 7: (A) Image taken using normal camera exposure control. Notice how the sky and the walls and windows are washed out. (B) Picture captured using a circle mask, masking the area outside of the lens, preserves the details.

Figure 8: When using omni-directional lens or as here a fish-eye lens the sky is more visible causing sun flares in the lens and difficult exposure situations.

manian ICT Centre is jointly funded by the Australian Government through the Intelligent Island Program and the CSIRO. The Intelligent Island Program is administered by the Tasmanian Department of Economic Development. The authors would like to thank the Autonomous Systems Laboratory team for their support of this work. Special thanks go to Stephen Nuske, Cédric Pradalier and Peter Hansen for their guidance and support.

References

[Canon, 2007] Canon. Canon technologies. Internet: http://www.canon.com/technology/, 2007.

[Corke *et al.*, 2004] Peter Corke, Pavan Sikka, Jonathan Roberts, and Elliot Duff. DDX: A Distributed Software Architecture for Robotic Systems. In *Proc. Australian Conf. Robotics and Automation*, Canberra, December 2004.

[Duff, 2005] Elliot Duff. DDXVideo: A Lightweight Video Framework for Autonomous Robotic Platforms. In *Proceedings of Australasian Conference on Robotics and Automation*, Sydney, Australia, December 2005.

[Horn, 1986] Berthold K. P. Horn. *Robot Vision.* The MIT Press, 1986.

[Intel Corporation, 2001] Intel Corporation. Open Source Computer Vision Libraby - Reference Manual. Intel Corporation, 2001.

[Jannerup and Soerensen, 2000] Ole Jannerup and Paul H. Soerensen. *Introduktion til Reguleringsteknink.* Polyteknisk Forlag, 2000.

[McHugh, 2007] Sean T. McHugh. Cambridge in colour. Internet: http://www.cambridgeincolour.com/, 2007.

[Nikon, 2007] Nikon. Nikon technologies. Internet: http://nikonimaging.com/global/technology/, 2007.

[Nuske *et al.*, 2006] S. Nuske, G. Wyeth, and J. Roberts. Extending the dynamic range of robotic vision. In *Proceedings of the IEEE International Conference on Robotics and Automation*, 2006.

[Reichmann, 2007] Michael Reichmann. The luminous landscape. Internet: http://www.luminous-landscape.com, 2007.

[Shirvaikar, 2004] Mukul V. Shirvaikar. An optimal measure for camera focus and exposure. In *Proceedings of the IEEE*, 2004.

[Unibrain, 2007] Unibrain. Unibrain fire-i. Internet: http://www.unibrain.com/, 2007.