Separable Linear Classifiers for Online Learning in Appearance Based Object Detection

Christian Bauckhage and John K. Tsotsos

Centre for Vision Research, York University, Toronto, ON, M3J 1P3
http://cs.yorku.ca/LAAV

Abstract. Online learning for object detection is an important requirement for many computer vision applications. In this paper, we present an iterative optimization algorithm that learns separable linear classifiers from a sample of positive and negative example images. We demonstrate that separability not only leads to rapid runtime behavior but enables very fast training. Experimental results underline that the approach even allows for real time online learning for tracking of articulated objects in real world environments.

1 Motivation and Scientific Context

A general trend in present day computer vision research appears to be the integration of machine learning techniques into visual processing. Especially in the case of object detection in real world environments, the entanglement of vision and learning has led to stunning results. Cascaded weak classifiers rapidly detect objects of constraint shape and texture [1]. Taking aim at varying shape and texture, recent contributions simultaneously learn lexica of salient object parts as well as global structures [2,3,4]. Cognitive approaches integrate reasoning and learning across and within several levels of processing [5].

Robust as they are, the above techniques all require extensive training times. This hampers their use in scenarios where online learning is mandatory, as in the case of vision systems that assist their users in real world tasks. Among the few current proposals for such a scenario is a system that applies the Winnow algorithm for learning linear classifiers to motion data [6]. Others propose the use of sequential principal component analysis (PCA) and probabilistic tracking [7], or apply VPL classification, a technique that combines vector quantization, PCA and locally linear maps [8]. However, although they are fast, none of these methods reaches real time performance in online learning for object recognition.

In this paper, we present a simple approach to very fast object learning which, nevertheless, provides rapid runtime behavior and reliable detection. Based on positive and negative example images, we propose an iterative least mean squares technique of learning separable linear classifiers. The method accomplishes input processing as rapidly as the popular cascaded weak classifiers. Moreover, it copes with objects of considerably varying shape and texture and is characterized by very short training times. Our classifiers therefore enable real time online learning in object recognition.

The next section first discusses the benefits of linear classifiers for visual object detection and then introduces our algorithm for learning separable classifiers. Section 3

A. Gagalowicz and W. Philips (Eds.): CAIP 2005, LNCS 3691, pp. 347–354, 2005.

presents experimental results in online learning for object detection. Finally, a discussion ends this contribution.

2 Separable Linear Classifiers for Object Detection

In their most basic form, binary linear classifiers compute the scalar product $\boldsymbol{w}^T\boldsymbol{x}$ of a parameter vector $\boldsymbol{w}$ and a feature vector $\boldsymbol{x}$. Their appeal for visual object detection lies in the fact that they may be implemented as two-dimensional linear filters. This requires writing parameters and features as matrices $\boldsymbol{W}$ and $\boldsymbol{X}$ and considering the Frobenius product of matrices $\boldsymbol{W} \star \boldsymbol{X} = \sum_{i,j} W_{ij}X_{ij}$. If $\boldsymbol{X}$ denotes a digital image and $\boldsymbol{W}$ a suitable finite impulse response filter matrix of size $m \times n$, a label y_{ij} characterizing the visual content in the vicinity of each pixel (i, j) can be computed from the convolution $\boldsymbol{W} * \boldsymbol{X}$

$$y_{ij} = \sum_{k=-m/2}^{m/2} \sum_{l=-n/2}^{n/2} W_{m-k,n-l} X_{i-k,j-l} = \boldsymbol{W} \star \boldsymbol{X}_{ij} \tag{1}$$

where $\boldsymbol{X}_{ij}$ denotes an image patch of size $m \times n$ centered at (i, j).

Note that if $\boldsymbol{W}$ is a $m \times n$ matrix, convolution requires $O(mn)$ operations per pixel. This may result in prohibitive computational costs even if m and n are set to moderate values. However, if $\boldsymbol{W}$ was a separable matrix, i.e. $\boldsymbol{W} = \boldsymbol{u}\boldsymbol{v}^T$ where $\boldsymbol{u} \in \mathbb{R}^m$ and $\boldsymbol{v} \in \mathbb{R}^n$, the two-dimensional convolution could be computed as a sequence of two one-dimensional convolutions $\big(\boldsymbol{X} * \boldsymbol{u}\big) * \boldsymbol{v}^T$. This would reduce the effort to $O(m+n)$ and therefore provide a fast linear approach to object detection. Next, we discuss how to obtain separable classifiers from training examples.

2.1 Iterative Least Mean Square Learning of Separable Classifiers

Our approach to classifier training modifies an algorithm introduced by Venkatachalam and Aravena [9]. In contrast to their work, we consider spatial convolutions instead of frequency domain filter design. However, for proofs of some of the assumptions applied below, the reader is referred to [9].

A convenient approach to binary classifier training applies the well-known method of least mean squares (LMS) optimization. If we require the parameter matrix to be a *one term separable filter* $\boldsymbol{W} = \boldsymbol{u}\boldsymbol{v}^T$ and if we consider the equivalence

$$\boldsymbol{u}\boldsymbol{v}^T \star \boldsymbol{X} = \sum_{k,l} (\boldsymbol{u}\boldsymbol{v}^T)_{kl} X_{kl} = \sum_{k,l} u_k v_l X_{kl} = \boldsymbol{u}^T \boldsymbol{X} \boldsymbol{v}, \tag{2}$$

then we can write the LMS error function as:

$$E(\boldsymbol{u}, \boldsymbol{v}) = \frac{1}{2} \sum_{\alpha} \big(y^\alpha - \boldsymbol{u}^T \boldsymbol{X}^\alpha \boldsymbol{v}\big)^2 \tag{3}$$

where $\{\boldsymbol{X}^\alpha, y^\alpha\}_{\alpha=1,\ldots,N}$ is a sample of image patches of size $m \times n$ with corresponding class labels. A solution for $\boldsymbol{u}$ and $\boldsymbol{v}$ can be determined as follows: Given an arbitrary vector $\boldsymbol{u} \in \mathbb{R}^m$, we can compute $\boldsymbol{x}_u^{\alpha^T} = \boldsymbol{u}^T \boldsymbol{X}^\alpha$ and then rewrite equation (3):

$$E(\boldsymbol{u}, \boldsymbol{v}) = \frac{1}{2} \sum_{\alpha} \left(y^{\alpha} - \boldsymbol{x}_u^{\alpha^T} \boldsymbol{v} \right)^2 \tag{4}$$

which is of the form usually encountered in LMS optimization. Consequently, we can determine the optimal set of weights $\boldsymbol{v}^*(\boldsymbol{u})$ by means of the usual approach of setting $\nabla_{\boldsymbol{v}} E(\boldsymbol{u}, \boldsymbol{v}) = 0$. A closed form solution for $\boldsymbol{v}^*$ exists if the correlation matrix $\boldsymbol{C} = \sum_{\alpha} \boldsymbol{x}_u^{\alpha} \boldsymbol{x}_u^{\alpha^T}$ is non singular. In this case, $\boldsymbol{v}^* = \boldsymbol{C}^{-1} \tilde{\boldsymbol{y}}$ where $\tilde{\boldsymbol{y}}$ denotes the cross correlation vector between inputs and labels.

Given $\boldsymbol{v}^*$, we can compute $\nabla_{\boldsymbol{u}} E(\boldsymbol{u}, \boldsymbol{v}^*)$ and hence determine $\boldsymbol{u}^*$. As we started with an arbitrary $\boldsymbol{u}$, these two steps have to be iterated until a convergence criterion is met. It can be shown that the solution does not depend on the length of $\boldsymbol{u}$. Therefore, we constrain the vector $\boldsymbol{u}$ to be of unit length $\|\boldsymbol{u}\| = 1$. On the one hand, this introduces additional effort, because it requires normalizing $\boldsymbol{u}$ after each iteration. On the other hand, as $E(\boldsymbol{u}, \boldsymbol{v}^*)$ becomes a continuous, convex function over the unit ball in $\mathbb{R}^m$, which is a compact set, normalization guarantees the convergence of the procedure. Moreover, the unit length constraint provides a simple convergence criterion. In our implementation, we use $\|\boldsymbol{u}_t - \boldsymbol{u}_{t-1}\| \leq \epsilon$, which proved to converge quickly.

for $i = 1, \ldots, k$
  randomly initialize $\boldsymbol{u}_i$
  normalize $\boldsymbol{u}_i \leftarrow \frac{\boldsymbol{u}_i}{\|\boldsymbol{u}_i\|}$
  orthogonalize $\boldsymbol{u}_i \leftarrow \boldsymbol{u}_i - \sum_{j=1}^{i-1} \frac{\boldsymbol{u}_j^T \boldsymbol{u}_i}{\boldsymbol{u}_j^T \boldsymbol{u}_j} \boldsymbol{u}_j$
  repeat
    $\boldsymbol{u}_i^{\text{old}} \leftarrow \boldsymbol{u}_i$
    solve $E(\boldsymbol{u}_i, \boldsymbol{v}_i) = \frac{1}{2} \sum_{\alpha} \left(y^{\alpha} - \boldsymbol{u}_i^T \boldsymbol{X}^{\alpha} \boldsymbol{v}_i \right)^2$ for $\boldsymbol{v}_i$
    orthogonalize $\boldsymbol{v}_i \leftarrow \boldsymbol{v}_i - \sum_{j=1}^{i-1} \frac{\boldsymbol{v}_j^T \boldsymbol{v}_i}{\boldsymbol{v}_j^T \boldsymbol{v}_j} \boldsymbol{v}_j$
    solve $E(\boldsymbol{u}_i, \boldsymbol{v}_i) = \frac{1}{2} \sum_{\alpha} \left(y^{\alpha} - \boldsymbol{u}_i^T \boldsymbol{X}^{\alpha} \boldsymbol{v}_i \right)^2$ for $\boldsymbol{u}_i$
    normalize $\boldsymbol{u}_i \leftarrow \frac{\boldsymbol{u}_i}{\|\boldsymbol{u}_i\|}$
    orthogonalize $\boldsymbol{u}_i \leftarrow \boldsymbol{u}_i - \sum_{j=1}^{i-1} \frac{\boldsymbol{u}_j^T \boldsymbol{u}_i}{\boldsymbol{u}_j^T \boldsymbol{u}_j} \boldsymbol{u}_j$
  until $\|\boldsymbol{u}_i^{\text{old}} - \boldsymbol{u}_i\| \leq \epsilon$
endfor

Fig. 1. Iterative algorithm to learn the parameter vectors $\boldsymbol{u}_i$ and $\boldsymbol{v}_i$ of a k term separable linear classifier

Note that the classifier that results from this procedure only comes along with $m+n$ coefficients, whereas, in the non-separable case, one would have learned $m \cdot n$ parameters. In its current form, the separable classifier therefore seems less flexible than the usual solution. However, having derived a solution for a one-term separable classifier, we can construct a separable classifier such that its weight matrix is the sum of k rank 1 separable matrices: $\boldsymbol{W} = \sum_{i=1}^{k} \boldsymbol{u}_i \boldsymbol{v}_i^T$.

One can show that, if $\boldsymbol{W}_k = \sum_{i=1}^{k} \boldsymbol{u}_i \boldsymbol{v}_i^T$ is a *k-term* representation of the coefficient matrix of a classifier, a $(k+1)$-term solution can be found by minimizing $E(\boldsymbol{u}_{k+1}, \boldsymbol{v}_{k+1})$, if, for $i \neq j$, the parameter vectors obey $\boldsymbol{u}_i^T \boldsymbol{u}_j = 0$ and $\boldsymbol{v}_i^T \boldsymbol{v}_j = 0$.

Given the one-term solution, a binary classifier with an arbitrary $k > 1$ can be generated using recursion. The above optimization method simply has to be extended, such that, after *each* iteration, the vectors $\boldsymbol{v}_{k+1}$ and $\boldsymbol{u}_{k+1}$ are orthogonalized with respect to $\{\boldsymbol{v}_i\}_{i=1,\ldots,k}$ and $\{\boldsymbol{u}_i\}_{i=1,\ldots,k}$, respectively. Orthogonalization can be done by applying the Gram-Schmidt procedure.

Figure 1 summarizes the iterative algorithm for training a k-term separable linear classifier. Next, we point out favorable characteristics of this approach, and then, we present results obtained with our separable object detectors.

2.2 Benefits of the Separable Approach

It is interesting to note that any matrix $\boldsymbol{W}$ can be written as a sum of separable matrices. This immediately results from the singular value decomposition $\boldsymbol{W} = \sum_{i=1}^{r} \sigma_i \boldsymbol{u}_i \boldsymbol{v}_i^T$, where r is the rank of $\boldsymbol{W}$, the σ_i are its singular values, and $\boldsymbol{u}_i$ and $\boldsymbol{u}_i$ denote the left and right singular vectors, respectively. Although the proof is omitted here, the orthogonality of the singular vectors actually establishes why, in our iterative approach, we must orthogonalize the coefficient vectors. However, this analytical, SVD-based approach to classifier design has little appeal for practical application.

For a separable classifier resulting from the SVD of a given coefficient matrix, convolving an image will require $O(r(m+n))$ operations per pixel. As the gain in speed depends on the rank r of $\boldsymbol{W}$, there will be no speed benefit in many practical cases. Dealing with the detection of elongated objects (see Fig. 2), $\boldsymbol{W}$ will be of rectangular form so that its rank will most likely be $r = \min\{m, n\}$. If, w.l.o.g, $r = m$, the convolution effort will amount to $m(m+n) = m^2 + mn > mn$ and separated classification will be even more expensive. Of course, the number of terms in the SVD representation of $\boldsymbol{W}$ can be reduced to $k < r$. However, although SVD yields the minimal Frobenius norm $\|\boldsymbol{W} - \sum_{i=1}^{k} \sigma_i \boldsymbol{u}_i \boldsymbol{v}_i^T\|_F$ for any k, practical experience shows that the corresponding classifiers perform worse than the original one. Using our learning algorithm, both of these drawbacks can be avoided. As the separable classifier is derived directly from data rather than from the optimal non-separable version, our algorithm guarantees reasonable results, even in the case where $k \ll r = rank(\boldsymbol{W})$. Of course, choosing a small k results in classifiers having fast runtimes.

Furthermore, the SVD approach requires knowledge of the $m \times n$ coefficient matrix $\boldsymbol{W}$ of a given classifier. Training this classifier using least mean squares requires the computation and inversion of a covariance matrix $\boldsymbol{C}$ with dimensions $mn \times mn$. For larger values of m and n and many training examples $\alpha = 1, \ldots, N$, this will be very time consuming—even on modern computers. Therefore, in addition to the speed benefit during runtime, our technique also significantly speeds up the training phase: The covariance matrices $\boldsymbol{C}_u$ and $\boldsymbol{C}_v$ that appear in the learning algorithm for separable classifiers are of considerably reduced sizes $m \times m$ and $n \times n$, respectively.

Finally, note that fast training and operation times allow us to consider fairly large values for m and n. The resulting linear classifiers thus can process data from very high dimensional feature spaces. This actually guarantees reasonable reliability in object detection. According to Cover's theorem [10], the probability of finding a suitable hyperplane that separates data of any class distribution increases with the dimension of

Fig. 2. Exemplary detection results on the *Coke* sequence provided by Black and Jepson [11]

Fig. 3. Exemplary detection results on the desk sequence provided by Gorges et al. [12]

the embedding space. Even if our approach only considers linear discriminance between classes, it can generally be expected to yield good performance.

The experimental results presented in the next section stress that these properties of separable linear classifiers provide an auspicious avenue to online learning for object detection.

3 Experiments in Online Object Learning

In our experiments, we considered several video sequences known from the literature on tracking or scene reconstruction. All sequences show various moving objects in real-world office environments.

In each experiment, the intended object was manually specified in the first frame of a sequence. Then, 30 image patches were randomly selected from the neighborhood of the object and were used as positive training examples (class label +1); 240 image patches randomly selected from outside the neighborhood served as counter examples (class label -1). Training and classification were carried out on simple grey-value intensity information. The activation threshold θ was set to the minimum value resulting from projecting the positive examples onto the normal of the hyperplane learned in the training phase.

After training on the first frame of a sequence, the subsequent frames were convolved with the resulting classifiers. This was done in a brute force manner, where the whole image was convolved and not merely the regions of interest. The intended object was said to be detected where the resulting filter response exceeded the activation threshold θ. Note that we applied a non-maximum suppression to the response map, which reduced the number of false positives. After λ frames, the classifier was retrained using the current image; we experimented with $\lambda \in \{3, 6, 9, \ldots, 30\}$.

Next, we discuss our findings for two of our test cases in more detail.

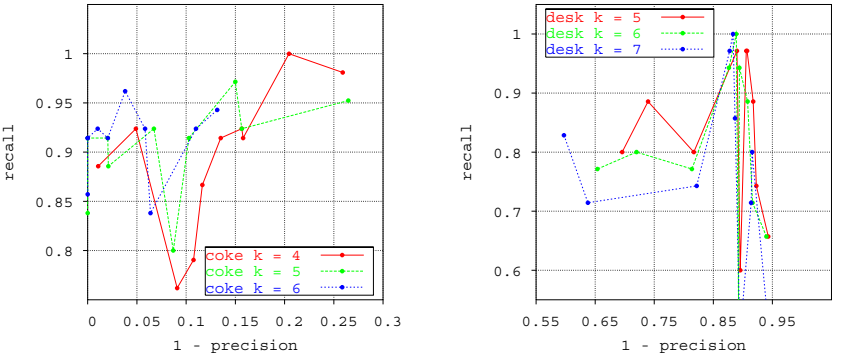

Fig. 4. Precision recall curves for the Coke sequence (left) and the desk sequence (right). The diagrams show the performance of different k term separable classifiers; the free parameter that was varied to generate the curves was the online update rate λ.

3.1 Coke Sequence

In the Coke sequence recorded by Black and Jepson [11], a tin can is moved in front of a static camera (see Fig. 2). We used 115×59 images patches to learn an appearance based model of the can. The diagram on the left in Fig. 4 shows precision recall curves obtained for separable classifiers of different ranks k. The highest recall (100%) resulted from a rank 4 classifier retrained every 21 frames. This classifier detected every instance of the moving can, however, the rate of false positives was 20%. The best performance was reached by a rank 6 classifier that was also retrained every 21 frames. Note that we measure performance quality in terms of equal error rate (EER), which characterizes the point where recall and precision are equal; for our best performing classifier we obtained an EER of 96%. Therefore, updating the classifier every 21 frames best captures the appearance variation due the the can's movement. The poorer performance for more frequent retraining appears to be an over-fitting phenomenon.

The diagram on the left in Fig. 5 plots recall and precision against operation frequency of the tested classifiers. As one would expect, the 4-term classifiers perform fastest. On a 3GHz Intel Xenon PC, the 155 frames, each of size 320×240, were processed at a frequency of approximately 10.5Hz, including file I/O *and* retraining. The most reliable 6-term classifier was measured to operate at 7.4Hz.

3.2 Desk Sequence

Though the results obtained on the Coke sequence are encouraging and representative for most of our experiments, a caveat remains concerning the feature space we considered for classification. The 86 frames of a resolution of 640×480 of the desk sequence provided by Gorges et al. [12] were recorded by a mobile camera panning across an office desk (see Fig. 3). To detect the CD on the left of the scene, we used 121×121 windows. Since the image sizes are four times as large as those in the Coke sequence, the operation frequencies of the resulting classifiers dropped to about a fourth of the

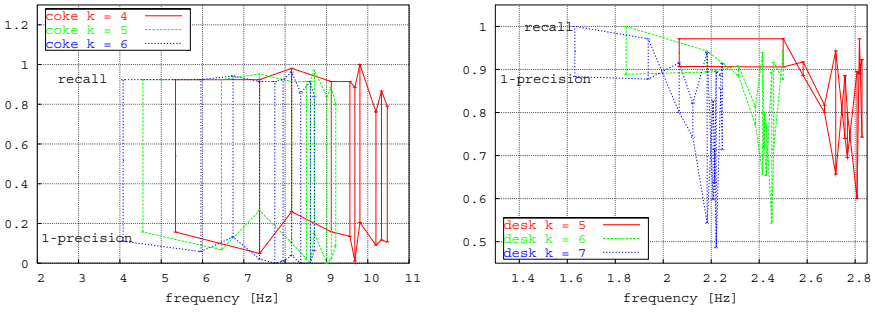

Fig. 5. Precision and recall and corresponding operation frequencies measured for the 320×240 images of the Coke sequence (left) and for the 640×480 images of the desk sequence (right). Again, the performance of different k term separable classifiers is shown for varying online update rates λ.

ones in the experiments above (see right hand side of Fig. 5). As shown in the digram on the right in Fig. 4, for all rank k classifiers considered in our tests, an update rate λ could be found that yielded high recall rates and reliable CD detection. However, the problem with this sequence is that there are many false positives because the cover of the magazine that is located behind the box becomes visible in the middle of the sequence. As it is very similar to shape and color of the intended object, the cover was frequently classified to depict the CD. The exemplary results in Fig. 3 were obtained with a 7 term separable classifier that was retrained every 18 frames and showed a recall of 83% and a false positive rate of about 59%.

Obviously, online learning did not provide a remedy in this pathological case. However, the problems encountered here do not constitute an inherent shortcoming of our technique. Rather, they are due to the type of features we considered in our experiments. Although simple intensity values yielded satisfactory results in many cases, the desk sequence shows that, depending on the application scenario, other features might have to be considered. To further improve our results, we are currently experimenting with an image representation framework recently proposed by Koenderink [13].

4 Conclusion and Outlook

This paper presented a fast and conceptually simple approach to visual object detection. We described an iterative, two-step optimization method for learning a binary, one-term separable linear classifier from a set of positive and negative example images. Given the optimal one-term classifier, classifiers of arbitrary higher ranks can be obtained from a recursive scheme. By design, the resulting classifiers correspond to linear filters. The classification process itself thus consists of convolving an input image. Since the detectors are separable, their runtime is fast—even for large filter masks. Nevertheless, the required training times are very short. While other recent contributions dealing with online learning report update times of slightly more than a second [6,7,8], our approach

reaches several Hz. Experimental results on image sequences known from literature show that this allows for real online learning and adaption of the classifiers.

There are numerous promising directions for further research on our rapid appearance-based approach to object learning. We will especially consider the following ideas: it would be interesting to see if separable classifier matrices can be subjected to affine transformations in order to reduce online update rates but nevertheless detect objects while the camera zooms or rotates. Moreover, integrating separable classifiers into tracking applications seems auspicious. One can imagine a particle filter for robust tracking, where the object models are given as separable classifiers. The adaption step of the filter would correspond to classifier retraining as in our current implementation. Also, if, in the verification step of the particle filter, the classifiers are only applied to small areas of the whole image, the resulting tracker should be fast and accurate.

References

1. Viola, P., Jones, M.: Rapid Object Detection using a Boosted Cascade of Simple Features. In: Proc. CVPR. Volume I. (2001) 511–518
2. Agarwal, S., Awan, A., Roth, D.: Learning to detect objects in images via a sparse, part-based representation. IEEE Trans. on Pattern Analysis and Machine Intelligence **26** (2004) 1475–1490
3. Fergus, R., Perona, P., Zisserman, A.: Object class recognition by unsupervised scale-invariant learning. In: Proc. CVPR. Volume II. (2003) 264–272
4. Leibe, B., Leonardis, A., Schiele, B.: Combined object categorization and segmentation with an implicit shape model. In: Proc. ECCV Workshop on Statistical Learning in Computer Vision, Prague (2004)
5. Bauckhage, C., Hanheide, M., Wrede, S., Sagerer, G.: A Cognitive Vision System for Action Recognition in Office Environments. In: Proc. CVPR. Volume II. (2004) 827–833
6. Nair, V., Clark, J.: An Unsupervised Online Learning Framework for Moving Object Detection. In: Proc. CVPR. Volume II. (2004) 317–324
7. Ross, D., Lim, J., Yang, M.H.: Adaptive Probabilistic Visual Tracking with Incremental Subspace Updat. In: Proc. ECCV. LNCS, Springer (2004) 470–482
8. Heidemann, G., Bekel, H., Bax, I., Ritter, H.: Interactive online learning. Pattern Recognition and Image Analysis **15** (2005) 55–58
9. Venkatachalam, V., Aravena, J.: Optimal Parallel 2-D FIR Digital Filter with Separable Terms. IEEE Trans. on Signal Processing **45** (1997) 1393–1369
10. Cover, T.: Geometrical and Statistical Properties of Systems of Linear Inequalities with Applications to Pattern Recognition. IEEE Trans. on Electronic Computers **14** (1965) 326–334
11. Black, M., Jepson, A.: Eigentracking: Robust matching and tracking of articulated objects using a view-based reprenstation. Int. J. of Computer Vision **26** (1998) 63–84
12. Gorges, N., Hanheide, M., Christmas, W., Bauckhage, C., Sagerer, G., Kittler, J.: Mosaics from Arbitrary Stereo Video Sequences. In: Pattern Recognition. Volume 3175 of LNCS., Springer (2004) 342–349
13. Koenderink, J.J., van Doorn, A.J.: Image Processing Done Right. In: Proc. ECCV. Volume 2350 of LNCS., Springer (2002) 158–172