

# Boosted Fractal Integral Paths for Object Detection

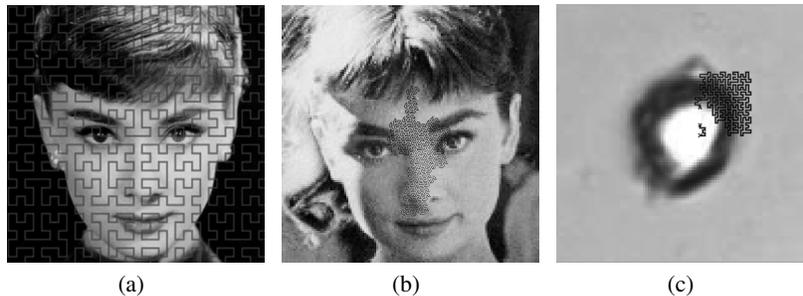
Arne Ehlers, Florian Baumann, Bodo Rosenhahn

Institut für Informationsverarbeitung (TNT)  
Leibniz Universität Hannover, Germany  
{last\_name}@tnt.uni-hannover.de

**Abstract.** In boosting-based object detectors, weak classifiers are often build on Haar-like features using conventional integral images. That approach leads to the utilization of simple rectangle-shaped structures which are only partial suitable for curved-shaped structures, as present in natural object classes such as faces. In this paper, we propose a new class of fractal features based on space-filling curves, a special type of fractals also known as Peano curves. Our method incorporates the new feature class by computing integral images along these curves. Therefore space-filling curves offer our proposed features to describe a wider variety of shapes including self-similar structures. By introducing two subtypes of fractal features, three-point and four-point features, we get a richer representation of curved and topology separated but correlated structures. We compare AdaBoost using conventional Haar-like features and our proposed fractal feature class in several experiments on the well-known MIT+CMU upright face test set and a microscopy cell test set.

## 1 Introduction

Most object detection frameworks inspired by Viola and Jones [1] are based on simple Haar-like features using integral-images. Haar-like features are easy and fast in computation, but represent only rectangular structures. Such detection frameworks often apply very large sets of low-resolution training images to learn the classifiers. These training sets mostly contain low-resolution object details which are often corrupted with compression artefacts. Hence features having rough shapes are sufficient to describe its



**Fig. 1.** (a) Illustration of the Peano-Hilbert curve traversing a face image. (b) Example of a selected fractal feature in training. (c) Microscopic cell with selected fractal feature.

gross characteristics. In contrast, we suggest to learn from high-resolution images to locate features that represent fine object details.

In this work, we propose and analyze a new integral image representation based on space-filling curves to explore fine non-rectangular structures. We introduce three types of fractal features based on the Peano-Hilbert-, Gosper- and E-Curve, respectively. Our new fractal feature class is evaluated on the well-established MIT+CMU upright face test sets A,C [2] and compared to standard Haar-like features. In addition to face detection we applied our fractal features to microscopic cell-data, see Figure 1(c) and 2. In Figure 1(a) is exemplary shown how the Peano-Hilbert fractal curve traverses an image plane. Figure 1(b) presents a feature formed by the Gosper curve and selected by AdaBoost for face detection.

### 1.1 Related Work

Object detection using classifiers learned by boosting algorithms is a popular topic in computer vision. In all components of such a detection framework, researchers are working to improve the detector’s performance. Zhang et al. [3] present a well structured survey of advances in face detection. They categorize developments into variations of the boosting learning algorithm and advances in feature extraction. The boosting algorithm has been improved by many researchers since Viola and Jones [1] introduced their object detection framework. E.g. FloatBoost [4] allows to drop earlier weak classifiers having a high classification error later in training. MILBoost [5] addresses the problems given by high variations in the training set that e.g. unlabeled subcategories induce. Hence cost functions known from Multiple Instance Learning are integrated into the boosting algorithm. Other variants, such as LPBoost, SoftBoost or S-Adaboost [6–8] control the detection performance by interventions in the learning and weighting scheme itself. Xiao et al. propose a very efficient method to dynamically learn a cascade of classifiers [9].

In the advances of feature extraction, many different types of features based on histograms [10], binary patterns [11] or edges [12] have been developed. But a notable field of research is also the improvement of Haar-like features. For instance, Lienhardt and Maydt introduced a set of rotated Haar-like features [13] to extend the feature space.



**Fig. 2.** Example images of the training dataset consisting of faces and non-faces as well as cells and non-cells or corrupted cells.

Pham et al. developed polygonal Haar-like features [14] to increase the variety of shapes that a feature can represent. In this way, most of the improved Haar-like features rely on combinations of multiple rectangular structures.

## 1.2 Contribution

In our method, we replace the conventional Haar-like features by a new class of fractal features that are able to adopt to curved-shaped structures. Conforming with Haar-like features, the fractal features utilize an intermediate image representation to allow for an efficient computation. Preceding the feature extraction an integral image traversing along a fractal curve is calculated. In that way only two memory references are required to represent complex fractal structures by computing the sum of pixel intensities covered by that structure. Similarly to Haar-like features, the difference between pixel intensities in two image regions builds a feature. Utilizing three points on the fractal curve, a feature can exploit two adjacent fractal structures. The end point of the first path segment is here as well the starting point of the second path segment. Four-point features represent non-cohesive image regions defining separated fractal path segments. To summarize, our contributions are:

- We developed a new fractal feature class.
- Three-point and four-point features allow for a richer representation.
- We evaluated our new method in the field of face and microscopic cell detection.

The paper is structured as follows. Section 2 presents the proposed fractal feature class. The utilized fractal curves are described and their properties and construction are explained. Experimental results on face detection and microscopic cell detection are given in Section 3 while Section 4 concludes the paper.

## 2 Boosted Fractal Integral Paths

Boosted fractal integral paths are based on the object detection framework developed by Viola and Jones [1]. They introduced AdaBoost, a machine learning algorithm proposed by Freund and Schapire [15, 16], into object detection. In that application, AdaBoost constructs a strong classifier for object detection as a linear combination of weak classifiers based on Haar-like features.

In this work, we propose to learn an object detector based on fractal features. We describe the new class of fractal features in detail in this section.

### 2.1 Fractals

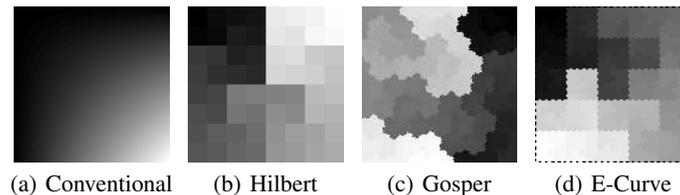
Following [17], a fractal is "a rough or fragmented geometric shape that can be split into parts, each of which is (at least approximately) a reduced-size copy of the whole". Such a property is also called self-similarity so that a pattern observed in one scale can often be found on other scales. There exist many examples in nature which demonstrate the beauty and importance, but also frequent appearance of fractals, e.g. in crystals, in snow flakes, or plants such as the romanesco broccoli. Therefore we assume that fractals can provide a good description for structures in all natural images including the test sets of faces and microscopic cells employed in our work.

## 2.2 Fractal Features

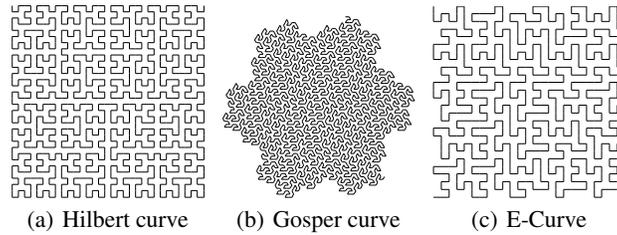
We want to take advantage of that common appearance of fractal structures by the feature model of our learned classifier. Hence, our proposed class of features utilizes a special type of fractal curves to compute fractal integral images along these curves. Conventional integral images are constructed by summing up pixel intensities from the upper left to the lower right corner of an image. Similarly, the fractal integral images integrate the pixel intensities of a 2D image plane along the fractal curve. Figure 3 displays the conventional integral image used for Haar-like features and the fractal integral images following three different fractal curves. These integral images have been normalized and computed on a homogeneous image. Precalculated integral images provide the benefit, that only few references into the integral image are required to compute the sum of pixel intensities in a region of the original image. In that way, pixel sums of arbitrary rectangular regions can be calculated by accessing four points in the conventional integral image. In contrast, differences between two pixels in the fractal integral image represent the sum of pixels covered by diverse fractal structures having a huge variability of shapes. According to Haar-like features, a fractal feature is as well calculated as the difference of the sums of pixel intensities in two image regions. These regions are defined by sampling numerous positions on the fractal integral path and build the feature set of the AdaBoost machine learning algorithm.

## 2.3 Fractal Properties

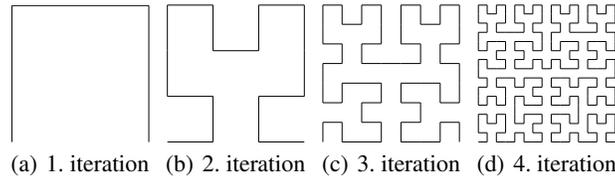
To appropriately construct the integral image it is desirable that the followed fractal path traverses every pixel in the image exactly once. This property is given by space-filling curves also referred to as Peano curves. One member of this type of fractal curves is the Peano-Hilbert curve. Like other space-filling curves it has the property of creating a 1D-representation of a 2D-image while preserving its proximity relationship better than a raster scan. Thus the Peano scan was examined for texture analysis and image compression due to its improved autocorrelation [18–21]. The integral image shown in Figure 3(b) illustrates the proximity of the Peano-Hilbert curve as each quadrant is completely traversed before the next quadrant is entered. Several space-filling curves are known but not all of them are suited for fractal integral paths. Fractals that base on tree structures as H tree fractals cannot be used. The difference of two pixels of an integral image computed on this tree is not as required in any case the sum of pixels of the original image along the fractal path that connects these two points. The Z-order curve



**Fig. 3.** Normalized integral images traversing different paths on a homogeneous image.



**Fig. 4.** Fractal curves used to traverse image plane and to define fractal structures exploited in feature computation.



**Fig. 5.** First four iterations of the Peano-Hilbert curve.

has applications similar to the Peano scan, but the Peano-Hilbert curve is preferred in this paper due to its better preservation of proximity. The E-Curve is favored over the Moore curve because its shape has compared to the Moore curve a stronger difference to the Peano-Hilbert curve. Space-filling curves that are closed curves like the Sierpinski curve are not suited as well. They could be splitted defining a start and an end point, but the property of the conventional integral image that the start and end point are at different sides of the image plane would be lost. We choose the Gosper curve as a third fractal for our method. Figure 4 illustrates the selected fractal curves. The Gosper curve is not a space-filling curve in terms of the definition given above as it traverses approximately every sixth pixel of the 2D-image a second or third time. But due to the locality of the curve this yields only some slightly enlarged features in which the corresponding pixels are weighted two or three times. As the feature selection process is performed by the AdaBoost algorithm with respect to the minimization of the classification error the disparity of those fractal features is acceptable. We select the Gosper curve because of its different shape containing angles that are multiples of 60 degrees. This leads also to an non-square outer boundary. Hence we clip an inner part of the curve as the fractal path has discontinuities at its boundary. Similar to the disparities in features this discontinuities are also tolerable due to the feature selection process.

## 2.4 Construction of Fractals

Fractal curves can be build using a Lindenmayer system, also referred to as L-System. The biologist Aristid Lindenmayer defined in 1968 a mathematical model to simulate the growth of multi-cellular organisms [22]. He developed a system of string replacement rules which are applied in parallel to recursively create an output string. The L-System grammar is defined by a tuple  $G = (V, \omega, P)$ , where

- $V$  is the alphabet of the system,
- $\omega$  is a string of symbols from  $V$  and defines the initial state,
- $P$  defines a set of production rules. Each rule consists of the predecessor, a string of symbols from  $V$ , and the successor, the string of symbols from  $V$  the predecessor is replaced by.

In contrast to a formal grammar the production rules of a L-System are in parallel applied in each iteration of the system. The alphabet  $V$  consists of constant symbols that are not substituted by the production rules and variables that are replaced and thus can be found on the left hand side of the rules. In our topic of constructing a small set of space-filling curves we use context-free L-Systems, in which the production rules only refer to an individual symbol and do not take its neighbouring symbols into account.

Interpreted as turtle graphics the output of a Lindenmayer system can be used to construct fractals. In turtle graphics a so-called turtle bot draws by executing a queue of simple instructions, like draw line, turn left and turn right. The length of one line segment and the angle to turn is often given as global parameter that can be a function of the iteration depth. Therefore each constant in the alphabet of the Lindenmayer system represents a command for the turtle bot.

The Peano-Hilbert curve for example can be described by the following L-System:

$$\begin{aligned}
 & - V = \{X, Y, +, -\}, \\
 & - \omega = X, \\
 & - P : \begin{cases} X \rightarrow +YF - XFX - FY+ \\ Y \rightarrow -XF + YFY + FX- \end{cases},
 \end{aligned}$$

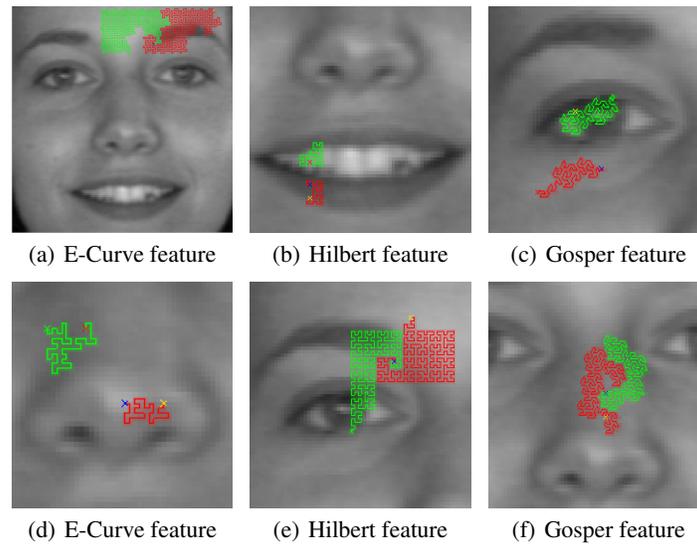
where  $F$  instructs the turtle bot to draw a line of length 1,  $+$  to rotate anticlockwise by  $90^\circ$  and  $-$  to rotate clockwise by  $90^\circ$ .  $X$  and  $Y$  are variables and thus do not represent commands to the bot. Figure 5 shows the first four iterations of the Peano-Hilbert curve starting at an initial angle of  $0^\circ$ . Similarly, the E-Curve and the Gosper curve can be constructed by slightly more complex L-Systems that are, due to the limited space, not reported in this paper.

## 2.5 Feature Types

To describe diverse structures we implemented two different types of features for each fractal, three- and four-point features. Their specific name refers to the property that the calculation of these fractal features requires only three and four memory references, respectively. Three-point features represent two adjacent integral path segments and thus give preferences to cohesive regions. Similarly the Haar-like features used by Viola and Jones only describe connected areas but require in the case of a two-rectangle feature six memory references. Additionally we define four-point features that describe separated regions that better conform to diverse structures.

## 3 Experimental Results

We applied fractal features to two different fields of computer vision: face detection and the detection of microscopic cells.



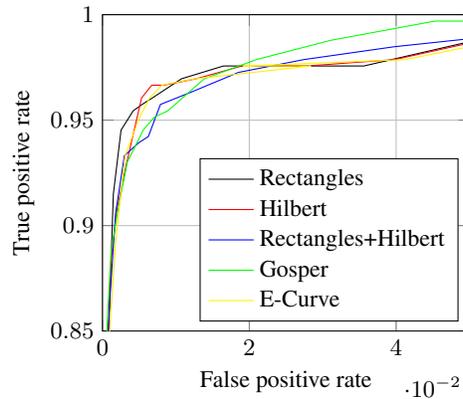
**Fig. 6.** Examples of fractal features found in training.

### 3.1 Face Detection

We trained our face detector on 1022 gray level images showing 340 individuals taken from the BioID face database [23], the MUCT face database [24] and the AT&T database of faces. Thus we give credit to AT&T Laboratories Cambridge. In an automatic process the faces were localized with respect to the given eye positions. The extracted images were then aligned and zoomed to a common scale resulting in a final patch resolution of  $128 \times 128$  pixels.

We trained our three types of fractal features and, for comparison, Haar-like features for 46 rounds. Figure 6 presents some fractal features selected in the training process. For validation we tested these boosted classifiers on the MIT+CMU frontal face dataset A and C [2]. Figure 7 presents the detection performance in a ROC curve. The fractal features and Haar-like features (Rectangles) show different characteristics. On the one hand the Haar-like features demonstrate better results in the high precision range and are at some point outperformed by the Hilbert fractals. On the other hand the fractal features achieve higher true positive rates.

In a second experiment we want to combine the characteristics of rectangle and fractal features. Hence we select the Hilbert fractals as they achieved the best results on our microscopic cell test set (see Table 1) and incorporate them into a combined rectangle-fractal framework. The training success of the combined framework is compared in Figure 8 to the corresponding homogeneous frameworks. The combined framework shows less fluctuations in the detection rates during training indicating that the different characteristics of fractal and rectangle features stabilize the combined training. Figure 8(d) demonstrates despite some fluctuations the overall improvement of the combined classifier compared to a pure Haar-like classifier.



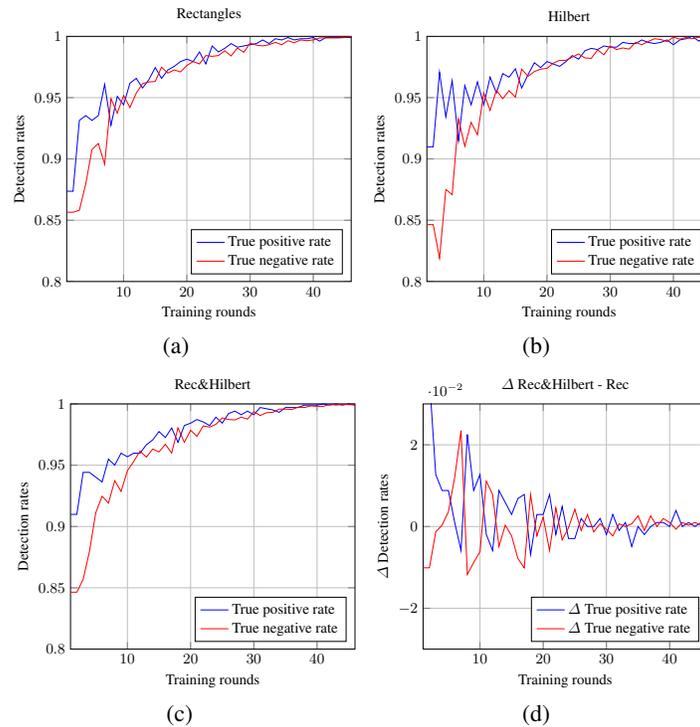
**Fig. 7.** ROC-Curves showing the detection performance on the MIT+CMU frontal face dataset. Classifiers consisting of fractal features are compared to conventional Haar-like features (Rectangles) and a mixed classifier of Peano-Hilbert fractals and Haar-like features.

We conducted additional experiments on degraded and modified versions of the training data to give an analysis of the strengths and weaknesses of the different feature classes. Figure 9(a) illustrates the detection rates achieved on the positive training set when its contrast is degraded. Controlled by a parameter between 0 and 1 the appearance of each training image is transformed between the original image having the expected contrast and a homogeneous mean image with minimal contrast, respectively. In Figure 9(b) the influence of rotation is shown. The training images are therefor rotated up to  $45^\circ$  prior to the detection process. It can be observed that especially in case of low contrast the application of fractal features can improve the detection performance. In case of strong rotations the rectangle features perform better. These observations indicate that fractal features fit closer to curve-shaped object structures. The closer fitting can result in an improved robustness to contrast changes but can also lead to a higher sensitivity to rotations.

The over-all performance of our tested detection framework is not as high compared to very sophisticated face detectors as we focused our studies on the impact of features and compared our new feature class to the conventional Haar-like features. Hence we intentionally selected a basic, non-cascaded boosting framework and relinquished additional pre- or post-processing steps like e.g. canny pruning in OpenCV [25] to increase the detector’s performance. Another reason for the performance gap is the discrepancy in the properties of our training set and the MIT+CMU dataset. We constructed and trained highly adaptable features suitable for higher image resolutions that are common these days. In contrast to that the well-known MIT+CMU test set contains several low-resolution images which do not provide fine details like our training set.

### 3.2 Microscopic Cell Detection

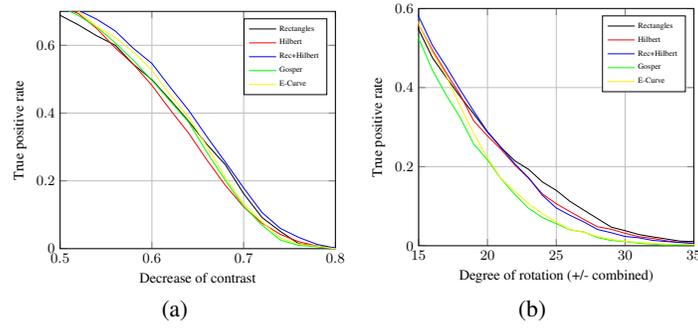
Additional experiments are conducted using microscopic cell images, see Figure 2. The cell images are acquired during cryo-conservation and as a result ice fronts are rising



**Fig. 8.** (a)-(c) Detection rates showing training success in face detection vs. amount of training rounds. (d) Difference in detection rates of Rectangles+Hilbert and Rectangles only vs. amount of rounds, presenting the benefit of mixed training.

around the cells. The goal is to detect (and track) the cells in videos. 250 images of cells and 350 images of non-cells are collected in a reasonable database. These image bases have been divided into a training and validation set using a 67/33 ratio. Crowther and Cox [26] illustrated that especially for small bases a split containing only a small part for validation is not recommendable. They suggested a ratio between 50/50 and 70/30. We trained classifiers for Haar-like, Hilbert-, Gosper- and E-curve features. Table 1 presents the results of classifiers trained in 50 rounds. The results illustrate that the fractal features and above all the Hilbert curve increases the detection performance described by the  $F_1$  score.

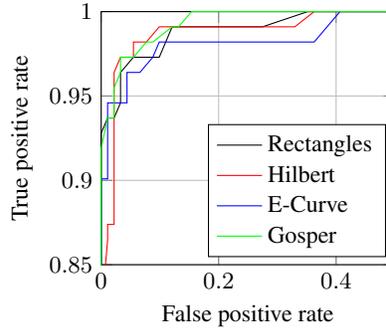
The ROC curve in Figure 10 illustrates that the fractal features, except for the E-curve, slightly outperform the Haar-like features. The Gosper curve clearly reaches first a TP-rate of 100%. The dependency of the detection rate from the number of training rounds is illustrated in Figure 11. In earlier rounds of the training the detection rates of the fractals (see Figures 11(b), 11(c), 11(d)) are more stable having less fluctuation. Figure 11(a) shows that the rectangles reach with less classifiers 100% TN-rate but the TP-rate decreases to 93,69%. But in the following rounds the detection rates of the rectangle features highly fluctuate.



**Fig. 9.** (a) Detection rate on training face images with lowered contrast. (b) Detection rate on rotated training face images. For visual clarity the intermediate interval, showing most differences, is presented in both figures.

**Table 1.** Results of microscopic cell-data detection. Hilbert curve achieves the best result.

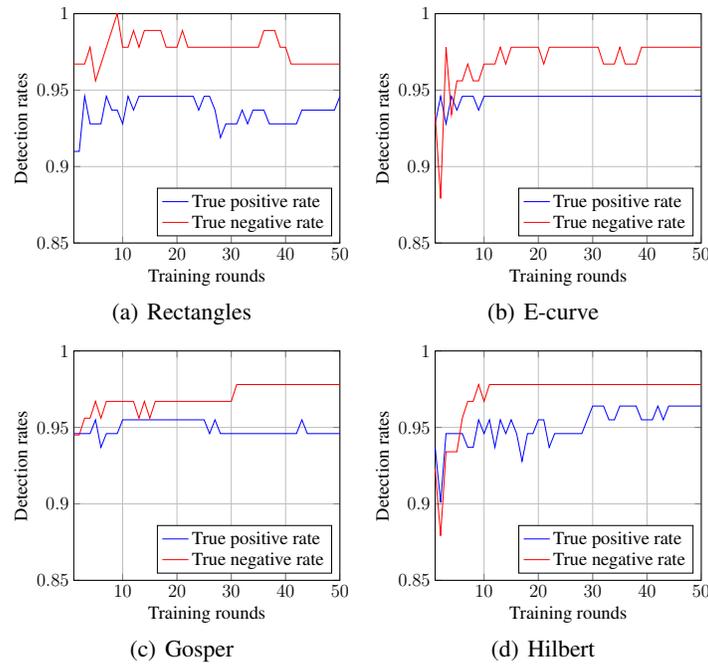
Application	Feature class	$F_1$ score
Microscopic cell-data	Rectangles	95.89%
Microscopic cell-data	Hilbert curve	97.27%
Microscopic cell-data	Gosper curve	96.33%
Microscopic cell-data	E-curve	96.33%



**Fig. 10.** ROC-Graph showing the detection performance on the microscopic cell dataset.

### 3.3 Training- and Computing Time

Due to precompiled fractal paths the training time between conventional- and fractal features does not differ. Furthermore the classical- and fractal integral image computation is an initial process at the beginning of the algorithm. There are some differences in the validation process using a sliding window. For the classical Haar-like features it is sufficient to compute the integral image once for each scale and cut out sub-windows at arbitrary positions. In contrast using fractal features we must compute the fractal integral image for every sub-window. Despite precompiled fractal paths the validation



**Fig. 11.** Detection rates showing training success in microscopic cell detection vs. amount of training rounds.

time, applying a sliding window, is slower. But this disadvantage could be overcome for example by programmable hardware like FPGAs.

## 4 Conclusions

We introduced a new type of features for object detection which describe fractal structures that enable to better adapt to curved-shaped objects. Our experiments in the domains of face detection and the detection of cells during cryo-conservation showed the improved detection performance of the fractal feature class.

Indeed, the usefulness of the fractal integral paths highly depend on the object classes to be detected. E.g. artificial objects, such as cars or manufactured parts might be better detected with rectangular features. But to our experience, especially for high-resolution images of natural object classes, the fractal curves lead to a noticeable improvement with only minor algorithmic modifications.

This seamless integration enables our approach to be easily utilizable in several boosting frameworks using integral image representations.

## References

1. Viola, P., Jones, M.J.: Robust real-time face detection. *International Journal of Computer Vision* **57** (2004) 137–154
2. Sung, K.K., Poggio, T., Rowley, H.A., Baluja, S., Kanade, T.: MIT+CMU frontal face dataset a, b and c. (MIT+CMU)
3. Zhang, C., Zhang, Z.: A survey of recent advances in face detection. Technical report, Tech. rep., Microsoft Research (2010)
4. Li, S.Z., Zhang, Z., Shum, H.Y., Zhang, H.: Floatboost learning for classification. *Advances in Neural Information Processing Systems*, Microsoft Research Asia (2002)
5. Viola, P., Platt, J.C., Zhang, C.: Multiple instance boosting for object detection. *Advances in Neural Information Processing* **18** (2007) 1417–1426
6. Warmuth, M.K., Gloer, K.A., Vishwanathan, S.: Entropy regularized l<sub>p</sub>boost. *Lecture Notes in Artificial Intelligence*, Springer-Verlag (2008)
7. Warmuth, M.K., Gloer, K., Raetsch, G.: Boosting algorithms for maximizing the soft margin. *Advances in Neural Information Processing Systems* MIT Press **20** (2008) 1585–1592
8. Jiang, J.L., Loe, K.F.: S-adaboost and pattern detection in complex environment. (CVPR 2003) 413–418
9. Xiao, R., Zhu, H., Sun, H., Tang, X.: Dynamic cascades for face detection. (ICCV 2007)
10. Wang, X., Han, T.X., Yan, S.: An hog-lbp human detector with partial occlusion handling. In: *Computer Vision, 2009 IEEE 12th International Conference on*, IEEE (2009) 32–39
11. Zhang, L., Chu, R., Xiang, S., Liao, S., Li, S.Z.: Face detection based on multi-block lbp representation. In: *Advances in biometrics*. Springer (2007) 11–18
12. Sabzmeydani, P., Mori, G.: Detecting pedestrians by learning shapelet features. In: *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*, IEEE (2007) 1–8
13. Lienhart, R., Maydt, J.: An extended set of haar-like features for rapid object detection. (IEEE ICIP 2002)
14. Pham, M.T., Gao, Y., Hoang, V.D.D., Cham, T.J.: Fast polygonal integration and its application in extending haar-like features to improve object detection. *CVPR 2010* (2010)
15. Freund, Y., Schapire, R.E.: A short introduction to boosting. *Journal of Japanese Society for Artificial Intelligence* **14** (1999) 771–780
16. Freund, Y., Schapire, R.E.: Experiments with a new boosting algorithm. *Proceedings of the Thirteenth International Conference* (1996) 148–156
17. Barnsley, M.F., Rising, H.: *Fractals Everywhere*. Boston: Academic Press Professional (1993)
18. Ansari, A.C., Fineberg, A.: Image data ordering and compression using peano scan and lot. *Consumer Electronics, 1992. Digest of Technical Papers. ICCE* (1992)
19. Dafner, R., Cohen-Or, D., Matias, Y.: Context-based space filling curves. *Comput. Graph. Forum* **19** (2000)
20. Lam, W.M., Shapiro, J.M., Sarnoff, D.: A class of fast algorithms for the peano-hilbert space-filling curve. In: *ICIP* (1). (1994) 638–641
21. Phuvan, S., Oh, T.K., Caviris, N.P., Li, Y., Szu, H.H.: Texture analysis by space-filling curves and one-dimensional haar wavelets. *Optical Engineering* 31(09) **31** (1992) 1899–1906
22. Lindenmayer, A.: Mathematical models for cellular interaction in development: Parts i and ii. *Journal of Theoretical Biology* **18** (1968) 280–315
23. BioID-AG: BioID face database. BioID Switzerland (2010)
24. Milborrow, S., Morkel, J., Nicolls, F.: The MUCT Landmarked Face Database. *Pattern Recognition Association of South Africa* (2010) <http://www.milbo.org/muct>.
25. Bradski, G.: The opencv library. *Dr. Dobb's Journal of Software Tools* (2000)
26. Crowther, P.S., Cox, R.J.: A method for optimal division of data sets for use in neural networks. Springer Berlin / Heidelberg, *Lecture Notes in Computer Science* **3684** (2005)