

# Interactive Images using Illustration Watermarks: Techniques, Studies, and Applications

Henry Sonnet  
O.-v.-Guericke-University  
Magdeburg, Germany  
sonnet@isg.cs.uni-magdeburg.de

Andrea Unger  
University of Rostock  
Rostock, Germany  
aunger@informatik.uni-rostock.de

Heidrun Schumann  
University of Rostock  
Rostock, Germany  
schumann@informatik.uni-rostock.de

## ABSTRACT

Static 2D images can be turned into interactive media by providing descriptive information that is shown on demand. For this purpose, two challenges need to be solved. First, appropriate techniques for the storage of descriptive information and its linkage to the image content are needed. We address this challenge by applying techniques from information hiding, a procedure we call *Illustration Watermarking*. In a user study, a new adaptive approach optimized for our application outperformed a traditional proceeding in terms of quality. The second challenge is to design interaction procedures which support the user in the exploration of descriptive information. Therefore, concepts are proposed to let the user find and retrieve the embedded data and its properties. Finally, we present various application scenarios.

**Keywords:** Image annotation, Information hiding, Data payload, Interactive images.

## 1 INTRODUCTION

The power of digital images to communicate information can be supplemented by additional descriptive information which becomes visible on the user's demand. For example, descriptions of what is depicted or additional knowledge about image content, origin, and author can be provided. Since the auxiliary information is only displayed on demand, a permanent occlusion of parts of the image content is avoided. Hence, static images are turned into media which can be interactively explored. Two aspects must be considered to realize the concept of interactive images: the storage of image and metadata on the one hand and interaction techniques to explore interactive images on the other hand.

To store images and their auxiliary data, we apply the concept of *Illustration Watermarking* introduced in [S<sup>+</sup>03]. It enables a compact storage of image and metadata within one file that can easily be distributed. Since the metadata is integrated into the image content, the file size does not depend on the volume of metadata and common file formats can be used. Thus, the image can still be accessed with common image viewers although special software might be required to explore the metadata. As another advantage, various metadata objects differing in content and type can be specified and linked to specific components of the image content.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.  
Copyright UNION Agency - Science Press, Plzen, Czech Republic.

The term *Illustration Watermarking* is chosen because watermarking refers to the invisible encoding of data within digital media while the data is related to the content of the media. However, the additional term *illustration* confines the approach from usual watermarking applications which provide copyright information or ensure data origin authenticity since this does not apply for our application scenario. In interactive images, the auxiliary information is commonly accessible and provides additional value for both the author and the user. Hence, aspects such as security and robustness are neglected. *Illustration Watermarking* focuses, instead, on techniques to embed a significant amount of data (*high capacity*). Further important aspects are to avoid perceivable modifications of the image content (*imperceptibility*), to enable watermark recovery without the original image (*blind detection*), and to embed the watermark message locally (*content-based*).

A second important topic besides storage is the design of user interaction techniques to explore interactive images, since the user should easily identify which image objects are augmented. Additional information, such as the type of metadata (e.g., texts or images), can guide the user to information he is interested in. For this purpose two concepts are proposed. The *Informative Cursor* extends the known mouse cursor with arrows that point to augmented image regions and encode properties of these regions. Based on the *Magic Lens* concept, the *Meta-Previewer* provides details about the metadata that is linked to a focused region as well as the locations of all augmented regions in the image.

In this paper, we demonstrate how *Illustration Watermarking* can be used to store descriptive information within a 2D image. After reviewing related work in Section 2, we describe a data storage technique which adapts the amount of embedded data to the image con-

tent and its evaluation in Section 3. In a user study, our technique outperformed a standard technique in terms of image quality. Aiming at an easy exploration of embedded metadata, *Informative Cursors* and the *Meta-Previewer* to localize augmented image regions and to retrieve properties of the metadata are introduced in Section 4. In Section 5, we present various application scenarios for *Illustration Watermarking*, before Section 6 concludes this paper.

## 2 RELATED WORK

The concept to augment images with additional data is not new. Existing solutions are reviewed in the following, separated into techniques for augmenting either digital images (Section 2.1) or printed images (Section 2.2). The concepts of information hiding and digital watermarking are shortly summarized in Section 2.3.

### 2.1 Storage of image and metadata

In this paragraph, techniques to store a digital image and its descriptive information are reviewed. Depending on whether the image and its metadata are included within the same data file or stored separately, we distinguish between integrated and separated data storage.

Regarding integrated data storage, software such as Macromedia Flash (e.g., [RD04]), Tool-Book (e.g., [SS04]), or MS PowerPoint (e.g., [Wem03]) can be employed to create highly complex interactive illustrations. The final illustration can then be stored as a single data file. However, displaying and interacting with those illustrations requires specific software.

A number of graphics format specifications provide a certain amount of space for metadata storage. The graphics file format JPEG2000, for example, allows to include metadata elements (XML boxes) in JPX files (e.g., [JTC04, JW04]). Other file formats provide private tags (TIFF, described in [Ass92]) or private text chunks (PNG, specified in [SW04]). In the medical domain, the DICOM standard [Com04] is widely-used for handling, storing, and transmitting patient-relevant data.

Storing metadata as a sequence of well-confined descriptions focuses on textual information. The Extensible Markup Language (XML) can be considered as the most popular standard underlying those descriptions. Examples are Scalable Vector Graphics (SVG) [Gro03] for handling 2D graphics and graphical applications using XML, or Extensible 3D (X3D) (e.g., [GC04]) as a 3D file format that facilitates delivering 3D data and other information. Although descriptive languages focus on words rather than pixel representations, even raster graphics can be included in descriptive representations (e.g., [AT04]). Since these approaches do not directly link image objects and their descriptions, a content-based interactive exploration of images can hardly be realized. Hence, our approach provides a close connection between image content and metadata.

Separated data storage refers to techniques that use individual data files or that represent image and metadata in database management systems. A simple example for handling individual data files are Web pages which include text and image data, and which were generated using Dynamic HTML (e.g., [Goo02]) or Synchronized Multimedia Integration Language (SMIL) [Gro05]. These languages do not only allow to structure the data appropriately, they also provide concepts for interaction. They include links that point to separately stored data files. Metadata can also be regarded as a collection of data items related to each other. Those data items and their interrelations can be managed by a database management system (DBMS) (e.g., [G<sup>+</sup>02, Hal01]). Employing DBMSs, the entire data can be efficiently processed. Examples for DBMSs are DB2, Oracle, or SQL Server.

However, to extract data from DBMS, appropriate queries have to be carried out. Our approach is to provide an intuitive image-based information searching.

### 2.2 Linking media with metadata

Our approach can be classified as digital-only, i.e., the cover medium as well as the watermark are virtual. However, several approaches exist that follow the same idea as ours, namely to enrich a medium with additional information, but are hybrid: they combine real-world media and virtual data. *Smart Images* [Ala00], for example, are either digital or real and contain visually imperceptible data such as Web pointers or contact information. Printouts of digital images which are augmented with metadata are distributed. After scanning, the encoded data can be extracted from the digital copy. *Intelligent Paper* [DC98] refers to a standard sheet of paper with marks imprinted which can be detected by specific pointers. When those pointers are connected to the Web, information specified by the publisher can be retrieved. *Paper++* is a similar approach (e.g., [NS03]). Different virtual layers arranged on a digital copy of a printed document include links to objects in a database. To query the database, a user points at a printed page with a wand device that reads the page number and the position from a grid of bar codes printed on the paper. An alternative to bar codes are *DataGlyphs* [Hec01]. These are patterns of small strokes whose orientations represent bits. To extract the data represented by those strokes, the printed document is recorded before the encoded information is virtually overlaid on the document.

### 2.3 Information hiding techniques

Digital watermarking is "the practice of hiding a message about an image, audio clip, video clip, or other work of media within that work itself" [C<sup>+</sup>02]. Since it is often related to security aspects which does not apply for our concept of *Illustration Watermarking*, we

want to refer to the more general term of information hiding [KP00]. Looking at information hiding for images, techniques are usually separated by the domain used for data embedding. A simple approach (introduced in [KM92] and [T<sup>+</sup>93]) is to replace the least significant bits in the spatial domain. While the approach allows to embed comparably large amounts of data without perceivable image modifications, it lacks in robustness. Therefore, various techniques have been derived to embed data in the frequency domain [CW01] or in the wavelet domain [MU01]. An overview on information hiding techniques is given in [Z<sup>+</sup>07].

Similar to our intended application, information hiding has already been used for the annotation of image objects. The technique presented in [B<sup>+</sup>02] aims at indexing images to be retrieved from databases. To this end, two types of watermarks are embedded, one contains segmentation data and the other one contains indexing data. The concept introduced in [R<sup>+</sup>04] focuses on providing metadata (short text labels) for specific image regions. After interactive region selection, the regions can be augmented with the metadata. In both approaches, the metadata contained within the image is limited to indexes or short text labels.

### 3 ILLUSTRATION WATERMARKING FOR 2D DIGITAL IMAGES

As our main idea to realize interactive images, an image is stored along with its metadata by applying techniques from information hiding. In this section, an adaptive technique based on the analysis of image features is introduced and evaluated in a user study. We focus on RGB color images with 24 bits (8 per color component) per pixel. Due to the requirements of high capacity, transparency, blind detection, and region-based encoding, we choose an approach based on data embedding in the pixel's least significant bits (LSB). A high capacity is expected since those bits of the image data are replaced which contribute least to the perceived image content. To extract the metadata from an image, the bits belonging to the watermark must be identified. As the number of bits replaced in the pixels' color components is fixed in the standard LSB approach, the message can be reconstructed if the encoded pixels, the size of the bit stream, and the data type are known. To confine a watermarked region, its outline pixels are marked by manipulating the least significant bit of the blue component. Bit stream size and data type are stored in a specific pixel at a prominent location within the region.

#### 3.1 Adaptive Data Embedding

The basic LSB approach performs watermarking independently from the image content. An equal number of bits is replaced in every pixel. But the capacity is not constant among pixels; it depends on the image fea-

tures. To maximize the capacity, the number of bits replaced in pixels should be adapted to the image content.

We introduce a *Capacity Map* to store the capacity of every pixel and every color component. To ensure that the encoded metadata can be extracted from the image, the capacity maps built from the image must be identical in the encoding and decoding process. Therefore, image data used to build the capacity map is separated from data modified in the encoding process. Our tests (see Section 3.2) have shown that it is a good choice to restrict data encoding to the 4 least significant bits of each color component. Thus, the 4 most significant bits maintain the original values and are suited for generating the capacity map. Due to this condition, the maximum capacity is 12 bits per pixel, 4 in every color component. In the following, we describe the analysis of texture and color to build an image's capacity map.

The texture of an image has a strong effect on the capacity. The more homogeneous the texture, the easier content modifications are visible. Hence, a homogeneity measure is required to estimate an image's capacity.

Droogenbroeck and Delvaux [VD02] introduced an approach for gray-scale images which computes the capacity from the entropy of small image blocks. Since in our approach only the 4 most significant bits of color components can be used, we adapted the algorithm to meet this requirement as described in the following.

For every small image block of size 8x8 pixels, the entropy value is computed from the 4 most significant bits of one color component. Given the probability of occurrence  $p_i = P(F = f_i)$  of each symbol  $f_i, i = 1, \dots, L$  which can occur, entropy is defined as

$$\varepsilon = - \sum_{i=1}^L p_i * \log p_i$$

The entropy value is compared to thresholds to define the capacity of a color component for all pixels in the image block. If the entropy is larger than half the maximum entropy, a capacity of 4 bits is assigned. Otherwise, the entropy is compared to a quarter of the maximum entropy to assign either 3 or 2 bits as the capacity. The thresholds are based on informal test results.

For homogeneous regions with all pixels having the same color value, the entropy is minimal because only one color value appears in the region. The entropy will be maximal if all possible values appear equally within the region - which indicates an inhomogeneous texture.

In homogeneous regions, the capacity computed from the texture is low. Nevertheless, even homogeneous regions can have varying capacities, depending on the region's color. However, the widely used RGB color space is not suited to determine the capacity since it does not correspond to human color perception. The distance between the colors before and after data embedding, given in RGB, cannot be interpreted as the



Figure 1: Image types in the study (a-c) and their capacity maps (d-f): For better representation, values in the capacity maps have been scaled to a range between 0 and 255, brighter values indicating higher capacities.

perceived distance between the colors. Therefore, the RGB color space is called non-uniform. To determine a pixel's capacity based on its color, uniform color spaces need to be applied, for example, CIELAB and CIELUV [Sto03]. These color spaces represent perceived distances with Euclidean distance values.

A color's capacity can be described as the maximum distance to another color in RGB while both colors are within a defined distance in the uniform color space and, thus, equally perceived. To define a color's capacity, the perceived distances to colors which are adjacent in the RGB color space are computed. The search starts with the closest colors in RGB. The RGB distance is increased until a differently perceived color is found. The maximum distance in RGB to an equally perceived color is taken as the color's capacity. Given the requirement of blind detection, only the 4 most significant bits of the color components can be used to compute the capacity. Hence, the same capacity is assigned to all colors sharing the 4 most significant bits, which are considered as a color class. We choose the smallest capacity of all colors within a class as its capacity.

To combine texture and color analysis, both capacity values are computed for every pixel's color components and the higher value is chosen as the capacity.

### 3.2 Evaluation

To evaluate the results of our adaptive technique, a user study was conducted with two main intentions. One was to determine the reliability of computed capacities

by comparing them to evaluated capacities. It was also examined whether the adaptive approach leads to higher capacities than the standard LSB technique.

#### 3.2.1 Design and implementation

During the study, participants were asked to detect watermarked regions in images which differed in image type and encoding technique. Besides, bit streams of different sizes were encoded to determine the maximum capacity for a constant image quality.

Three image types with different features were included in the study. The image type *GreenPattern* (Figure 1a) consists of large homogeneous regions with only two colors. The capacity (Figure 1d) results from color information. In contrast, the texture of the image type *Flowers* (Figure 1b) is very inhomogeneous. The computed capacity (Figure 1e) mainly results from texture information. The image type *Landscape* (Figure 1c) was included to evaluate whether adapting the number of replaced bits among pixels increases the overall capacity. Thus, watermarked regions spanned both homogeneous and inhomogeneous image regions.

One equally sized and shaped region was watermarked in every image. Bit streams of different sizes were encoded in regions with comparable computed capacities. To compare both techniques, identical locations were chosen for watermarking if the same data volume was encoded in one image type.

112 volunteers participated in the study which was conducted on a standard PC with a high definition TFT

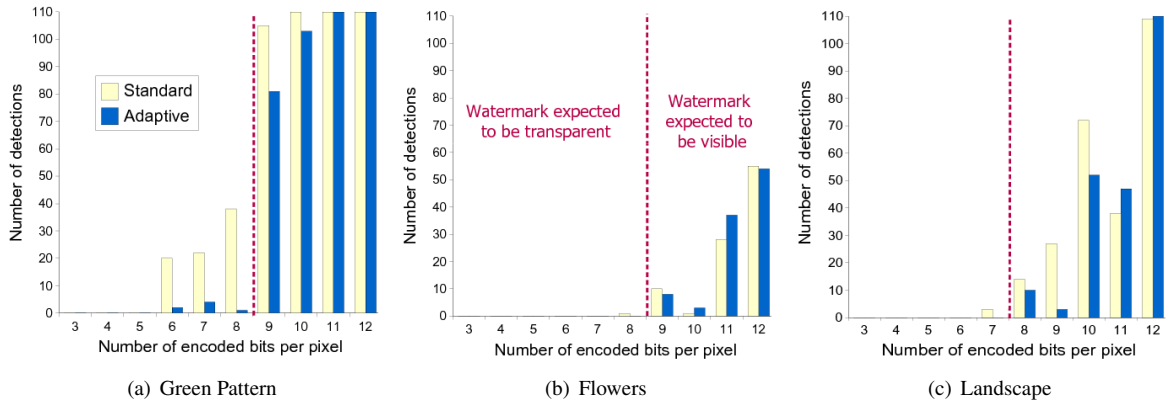


Figure 2: Results of the user study: For every image type, the number of participants who detected the watermark is shown for different volumes of encoded data (in bits per pixel). The standard LSB technique is shown in yellow bars, our adaptive approach in blue bars. The red dashed line indicates the expected visibility threshold.

flat screen. All images were presented for a maximum of 10 seconds in a random order, one at a time, with the three image types recurring periodically. The participants were instructed to mark the watermarked region with a mouse click which was internally determined to be within or outside the watermarked region.

### 3.2.2 Results

The computed capacities from our adaptive approach were evaluated based on the following assumption: For encoded data volumes smaller than the capacity, the watermark was expected to be transparent and otherwise visible. The results (Figure 2) show a coherence between the expected visibility and the number of detections (blue bars). For the image types *Landscape* and *Flowers*, none of the participants detected regions expected to be transparent. A few participants detected watermarks in the *GreenPattern* image with bit rates below the visibility threshold. However, the small number of detections is acceptable since our application aims at illustrative purposes. Thus, our capacity map is a well-found basis to encode data within an image without introducing perceptible content modifications.

Comparing the standard to our adaptive approach, a significant statistical difference in the number of detections was shown for *GreenPattern* and *Landscape*. Watermarks coded with the standard LSB method (yellow bars) were more often detected (*GreenPattern*: 520, *Landscape*: 263) than those coded with our adaptive method (*GreenPattern*: 415, *Landscape*: 222).

The adaptive approach has two advantages compared to the standard approach. First, the capacity of an image region is estimated. In an application scenario, the amount of embedded descriptive information can be limited to ensure transparency. Secondly, the adaptive distribution of the watermarking stream among the pixel's color components leads to a higher capacity especially if data is embedded in homogeneous regions, resulting in a capacity of 7 to 8 bits per pixel.

## 4 EXPLORATION OF METADATA

While in Section 3 technical aspects about metadata storage were discussed, the user interaction with the metadata is another important topic. First, the user needs hints about the - per default hidden - metadata contained in the image. After the user has requested metadata, an appropriate layout has to be applied. This second aspect is not covered in this work.

Considering the first aspect, visual cues are needed to communicate the existence and the properties of auxiliary data. The metadata should be easily detectable and context occlusion avoided. Also, a clear correlation between cue and augmented region is necessary. As important data properties, we consider location and size of regions being augmented, the kind of metadata that is used (text, image, sound) as well as the amount of data and the accordance to the user's interest. While most of these properties can be retrieved from the encoded data, we assume that the accordance to the user's interest is externally defined. We introduce two approaches to support the viewer in the exploration of interactive images, the *Informative Cursor* and the *Meta-Previewer*.

### 4.1 The Informative Cursor

The prototype of an *Informative Cursor* is the known mouse cursor which changes its style to indicate that a certain action can be performed. Beside showing the current position, the *Informative Cursor* utilizes arrows to lead the user to regions with associated metadata.

Graphical attributes encode the properties of the metadata, as shown in Figure 3. Two regions have been annotated. The first region, showing a daisy bloom, is associated with an image (indicated by the letter *I* in the arrowhead). The second region, showing a dandelion leaf, has corresponding text, as shown by the letter *T*. The lengths of the arrows indicate that the cursor is closer to region 1 in Figure 3a and closer to region 2 in Figure 3b. According to the thickness of



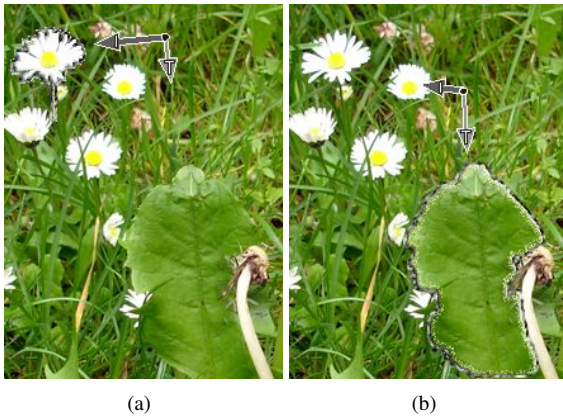


Figure 3: *Informative Cursors* used to provide information about annotated image regions. Boundaries of the closest annotated region have been emphasized.

the arrows, the amount of data associated with the first region is larger than in the second region. The darker color of the arrow pointing to the first region stands for a higher correlation with the user's interest. The closest annotated image region is emphasized by a highlighted contour, so the user can quickly reach the region and demand the metadata. Once the mouse cursor is over that region, it changes its shape to a standard cursor.

## 4.2 Meta-Previewer

The *Meta-Previewer* displays all annotated regions of an image by small markers and shows information about the metadata in a focused region. It follows the idea of *Magic Lenses* (e.g., [B<sup>+</sup>93]) because the *Meta-Previewer* is placed in front of the image to display additional information associated to the covered image region. It consists of a small rectangular shape which includes a down-scaled version of the original image with markers for all annotated image regions, the focused region is magnified by distortion (compare with *Fisheye Views*, e.g., [LA94, CM01, GS03] or *non-linear magnification* [KR97]). In addition, a frame provides further information about the metadata associated with the focused region.

Two examples are given in Figure 4. In Figure 4a, the mouse cursor is placed over the dandelion leaf in the lower image region. Within the *Meta-Previewer* located at the cursor position, the dandelion leaf is magnified, its contour highlighted. The small spheres used to mark all annotated image regions are visible. The right bar stands for the accordance to the user's interest. The small bars on top show the amount of metadata differentiated by type (*Text*, *Image*, *A* for combinations). For the focused region in Figure 4a, text is provided as metadata which correlates to the user's interest. In Figure 4b, the mouse cursor, and thus the *Meta-Previewer*, is moved to a second region. Here, text and image are

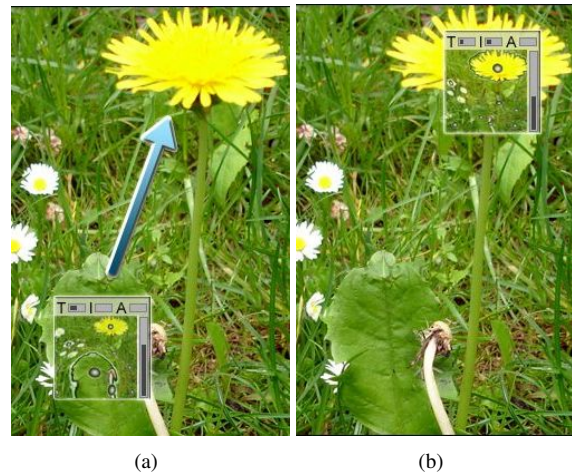


Figure 4: When the *Meta-Previewer* is moved across the image, its content is permanently updated. This includes magnifying the focused region as well as displaying information about its metadata.

given as metadata, but they correlate less with the user's interest than the metadata of the first focused region.

## 5 APPLICATION SCENARIOS

We see several application scenarios for *Illustration Watermarking*. Providing a content-based exploration of images, additional information such as texts and images can augment specific objects in the image. In Figure 5a, *Illustration Watermarking* allows to combine an image of a liver with a second image that provides a different view of the liver. Since the resulting image is still a standard raster graphic, the liver can be displayed using a standard image viewer. This can be useful when image previews are generated. Another example is given in Figure 5b. Here, photographs are linked to specific regions in a map to show what those places look like in real life. *Illustration Watermarking* can also be used for computer based learning. In Figure 5c, image objects have been characterized with appropriate terms. Using a built-in dictionary with translations of basic terms, the user can switch between object descriptions in different languages. The examples were generated with the software tool *Smage* (<http://www.smage.de>).

So far we have shown examples with static metadata which could either be displayed or not. We also see application scenarios with metadata content not only appearing on demand but also changing over time. All the information required for such application scenarios is stored as *Illustration Watermarks*. In addition to the components to be displayed, the metadata also contains information about how the content is to be animated and how it can be affected by the user. An example for a dynamic image is given in Figure 6. The dynamic components are small arrows that move along the paths blood takes through the heart. As two advan-



Figure 5: Example applications for *Illustration Watermarking*: In the image of a human liver (a), a second image (top) and text (bottom) are inserted into separated regions. A map (b) is augmented with photographs. In (c), the names of objects are shown in German and English.

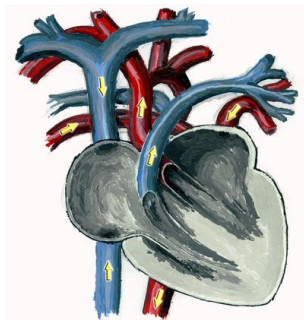


Figure 6: Heart illustration with moving arrows.

tages of animated arrows compared to static ones, they catch the user's attention easily and they characterize the exact paths better than static arrows would. The second example shown in Figure 7 is a simple game. The animated components are the apples and the arrow. The user can adjust the direction of the bow and trigger shots. The basic goal of the game is to hit each apple with the arrow before the apple comes off.

## 6 CONCLUSION

The concept of region-based *Illustration Watermarking* for 2D raster images has been proposed in this work. Two main challenges were addressed. First, a technique to store and retrieve descriptive information in an image has been described and evaluated. Approximately 6 bits per pixel can be embedded without decreasing the im-

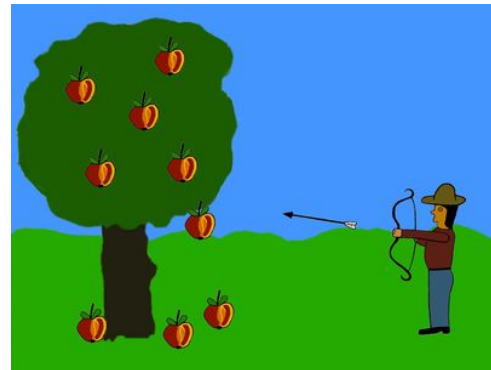


Figure 7: Image encoded with a simple game.

age quality. Depending on the image features, also 8 bits per pixel are feasible. Secondly, appropriate interaction techniques have been designed to let the user interactively explore the descriptive information. The concept offers a broad variety of application scenarios.

We focused on raster images to store descriptive information. While, our technique can be used for widely-used formats like PNG, it is not applicable to image formats based on lossy image compression, such as JPEG2000. Since techniques to store metadata content-based have already been developed for these formats [Leh07, K<sup>+</sup>06], we will extend our *Illustration Watermarking* framework to process images stored with lossy compression as well.

## 7 ACKNOWLEDGMENTS

The authors wish to thank Lothar Schlesier, Nguyen D. Quyen, and Sadiq Sowdagar for their implementation work.

## REFERENCES

- [Ala00] A. M. Alattar. Smart Images Using Digimarc's Watermarking Technology. In *Proc. of IS&T/SPIE's 12th International Symposium on Electronic Imaging*, volume 3971, pages 264–273, 2000.
- [Ass92] Adobe Developers Association. *TIFF 6.0 Specification*. 1992.
- [AT04] B. Antoniou and L. Tsoulos. Converting Raster Images to XML and SVG. In *Proc. of the 3rd Annual Conference on Scalable Vector Graphics (SVG Open 2004)*, 2004.
- [B<sup>+</sup>93] E. A. Bier et al. Toolglass and Magic Lenses: The See-Through Interface. In *Proc. of the 20th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH'93)*, pages 73–80. ACM Press, 1993.
- [B<sup>+</sup>02] N. V. Boulgouris et al. Segmentation and Content-Based Watermarking for Color Image and Image Region Indexing and Retrieval. *EURASIP Journal on Applied Signal Processing*, 4:420–433, 2002.
- [C<sup>+</sup>02] Ingemar Cox et al. *Digital Watermarking: Principles & Practice*. Academic Press, 2002.
- [CM01] M. S. T. Carpendale and C. Montagnese. A Framework for Unifying Presentation Space. In *Proc. of UIST 2001, ACM Symposium on User Interface Software and Technology*, pages 61–70. ACM Press, 2001.
- [Com04] DICOM Standards Committee. *Digital Imaging and Communications in Medicine (DICOM)*. DICOM PS 3 - 2004 standard, October 2004.
- [CW01] B. Chen and G. W. Wornell. Quantization Index Modulation Methods for Digital Watermarking and Information Embedding of Multimedia. *J. VLSI Signal Process. Syst.*, 27(1-2):7–33, 2001.
- [DC98] M. Dymetman and M. Copperman. Intelligent paper. In *Proceedings of the 7th International Conference on Electronic publishing*, pages 392–406, 1998.
- [G<sup>+</sup>02] M. Gertz et al. Annotating scientific images: A concept-based approach. In *Proc. of the 14th International Conference on Scientific and Statistical Database Management*, pages 59–68. IEEE Computer Society Press, 2002.
- [GC04] V. Geroimenko and C. Chen. *Visualizing Information Using SVG and X3D: XML-Based Technologies for the XML-Based Web*. Springer-Verlag, November 2004.
- [Goo02] D. Goodman. *Dynamic HTML: The Definitive Reference*. O'Reilly Media, Inc., September 2002.
- [Gro03] World Wide Web Consortium (W3C): SVG Working Group. *Scalable Vector Graphics (SVG) 1.1 Specification*. January 2003.
- [Gro05] World Wide Web Consortium (W3C): SYMM Working Group. *Synchronized Multimedia Integration Language (SMIL 2.0)*. 2nd edition, January 2005.
- [GS03] C. Gutwin and A. Skopik. Fisheye Views are Good for Large Steering Tasks. In *Proc. of the Conference on Human Factors in Computing Systems (CHI 2003)*, pages 201–208. ACM Press, 2003.
- [Hal01] T. Halpin. *Information Modeling and Relational Databases: From Conceptual Analysis to Logical Design*. Academic Press, 2001.
- [Hec01] D. L. Hecht. Printed Embedded Data Graphical User Interfaces. *IEEE Computer*, 34(3):47–55, 2001.
- [JTC04] Joint Technical Committee ISO/IEC JTC1, editor. *Information Technolog - JPEG 2000 Image Coding System: Extensions*. International Standard ISO/IEC 15444-2, 2004.
- [JW04] J. S. Janosky and R. W. Witthus. Using JPEG2000 for Enhanced Preservation and Web Access of Digital Archives - Case Study. In *IS&T's 2004 Archiving Conference*, volume 1, pages 145–149. IS&T - The Society for Imaging Science and Technology, April 2004.
- [K<sup>+</sup>06] G. Kazakeviciute et al. Self-annotating raster images. *Lithuanian journal "Information technology and control"*, 35(2):106–116, 2006.
- [KM92] Charles Kurak and John McHugh. A Cautionary Note on Image Downgrading. In *Proceedings of the Eighth Annual Computer Security Applications Conference*, San Antonio, TX, USA, 1992.
- [KP00] S. Katzenbeisser and F. A. P. Petitcolas. *Information hiding techniques for steganography and digital watermarking*. Artech House, Inc., 2000.
- [KR97] T. A. Keahey and E. L. Robertson. Nonlinear Magnification Fields. In *Proc. of the 1997 IEEE Symposium on Information Visualization*, pages 51–59. IEEE Computer Society Press, 1997.
- [LA94] Ying K. Leung and M. D. Apperley. A Review and Taxonomy of Distortion- Oriented Presentation Techniques. *ACM Transactions on Computer-Human Interaction*, 1(2):126–160, June 1994.
- [Leh07] C. Lehmann. Einbetten von Metadaten in JPEG2000-kodierte Bilder. Master's thesis, University of Rostock, 2007.
- [MU01] P. Meerwald and A. Uhl. A survey of wavelet-domain watermarking algorithms. In *SPIE Symposium, Electronic Imaging, Conference on Security and Watermarking of Multimedia Contents, San Jose, USA*, 2001.
- [NS03] M. C. Norrie and B. Signer. Web-based integration of printed and digital information. In *Proc. of the 2nd Workshop on Data Integration over the Web (DiWeb 02)*, volume 2590 of *Lecture Notes in Computer Science*, pages 200–219. Springer-Verlag, 2003.
- [R<sup>+</sup>04] Y. Rytzar et al. Interactive Segmentation with Hidden Object-Based Annotations: Toward Smart Media. In *Proc. of the SPIE, Storage and Retrieval Methods and Applications for Multimedia*, volume 5307, pages 29–37, 2004.
- [RD04] R. Reinhardt and S. Dowd. *Macromedia Flash MX 2004 Bible*. John Wiley & Sons, Ltd., 2004.
- [S<sup>+</sup>03] H. Sonnet et al. Illustration Watermarks for Vector Graphics. In *Proc. of Pacific Graphics*, 2003.
- [SS04] Inc. SumTotal Systems. *ToolBook Instructor 2004 - User Guide*. 2004.
- [Sto03] M. C. Stone. *A field guide to digital color*. A K Peters Ltd., 2003.
- [SW04] Joint Technical Committee ISO/IEC JTC1 SC24 and PNG Group (W3C). *Portable Network Graphics (PNG) Specification*. International Standard ISO/IEC 15948:2003, 2nd edition, March 2004.
- [T<sup>+</sup>93] A. Z. Tirkel et al. Electronic watermark. In *Digital Image Computing, Technology and Applications (DICTA'93)*, 1993.
- [VD02] M. Van Droogenbroeck and J. Delvaux. An entropy based technique for information embedding in images. In *IEEE Signal Processing Symposium*, pages 81–84, Leuven, Belgium, March 2002.
- [Wem03] F. Wempen. *Microsoft Office - PowerPoint 2003 Bible*. John Wiley & Sons, Ltd., 2003.
- [Z<sup>+</sup>07] Dong Zheng et al. A survey of RST invariant image watermarking algorithms. *ACM Comput. Surv.*, 39(2):5, 2007.