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

A static 2D image can be turned into an interactive medium 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 it's linkage to the image content are needed. We address this challenge by applying techniques from information hiding, a procedure we call *Illustration Watermarking*. Results of a user study show that a new adaptive approach optimized for our application outperforms 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 properties of the embedded data and the data itself. Finally, we present various application scenarios for *Illustration Watermarking*.

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, additional knowledge concerning the image content, or information about the origin and author of the image 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 media: the storage of image and metadata on the one hand and interaction techniques for the exploration of interactive images on the other hand.

To store images and their auxiliary data, we apply the concept of *Illustration Watermarking* introduced in [S^+03]. It enables a compact storage of image and metadata, referring to an integration within one file that can be easily distributed. Since the metadata is fused with 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.

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* shall confine the approach from usual applications of watermarking which provide copyright information or ensure data origin authenticity. 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 which can 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 beside storage is the design of user interaction techniques to explore interactive images. The user must be able to easily identify which objects in an image are augmented. Additional knowledge, for example what type of metadata is provided (like texts or images), can help to guide the user to information he is interested in. Therefore, two concepts are proposed. They do not only point to augmented image regions, but also provide knowledge about the properties of the metadata. *Informative Cursors* extend the known mouse cursor with arrows which point to augmented image regions and encode properties of these regions. Based on the idea of *Magic Lenses*, the *Meta-Previewer* provides detailed properties of the metadata 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. Results of a user study indicate that the proposed technique that adapts the amount of embedded data to the image content outperforms a standard technique in terms of image quality. Aiming at an easy exploration of embedded metadata, *Informative Cursors* and the *Meta-Previewer* are described to localize augmented image regions and retrieve properties of the metadata. Finally, we present various application scenarios for *Illustration Watermarking*.

This paper is organized as follows. After reviewing related work in Section 2, we describe a technique for data storage and its evaluation in Section 3. In Section 4, two concepts to explore interactive images are introduced. Applications scenarios are proposed in Section 5, before Section 6 concludes this paper.

2 RELATED WORK

Although information hiding techniques have not been used for this purpose so far, the concept to augment images with additional data is not new. A number of already existing solutions is reviewed in the following, separated into techniques for augmenting digital images (Section 2.1) and techniques which can be applied to printed images (Section 2.2). The concepts of information hiding and digital watermarking are explained in Section 2.3.

2.1 Storage of image and metadata

This paragraph addresses existing techniques which can be employed to store a digital image and its descriptive information.

Depending on whether the image and metadata are included within the same data file structure or stored separately, we distinguish between integrated data storage 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 interactive illustrations which are highly complex. 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.

A concept focused on textual information is to store metadata as a sequence of well-confined descriptions. The Extensible Markup Language (XML) can be considered as the most popular standard underlying those descriptions. Examples are Scalable Vector Graphics (SVG) [Gro03] as a specification 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 formed so that they can be included in descriptive representations (e. g., [AT04]).

Since these approaches record image content and image description separately, a content-based interactive exploration of raster images is hard to realize. Therefore, our approach provides a close connection between image content and metadata.

Separated Data Storage refers to storage techniques for illustrations whose layers are not included within a single data file. Beside handling individual data files, the image and its metadata can also be represented 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 not only allow to structure the data appropriately, they also provide concepts for interaction. They include links that point at data files which are stored separately.

Metadata can also be regarded as a collection of data items which are in relation to each other. Those data items and their interrelations can be managed by a database management system (DBMS) (e. g., [G^+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, there exist several approaches 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. ALATTAR, for example, discusses the concept of *Smart Images* [Ala00]. Those images 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 digitization (scanning), the encoded data can be extracted from the digital copy. DYMETMAN and COPPER-

MAN [DC98] present the concept of *Intelligent Paper*. Intelligent paper 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 barcodes printed on the paper. An alternative to barcodes are *DataGlyphs* introduced by Hecht [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^+02]. Since it is usually related to security aspects which does not apply for our concept of *Illustration Watermarking*, we want to refer to more general term of information hiding [KP00]. Looking at information hiding techniques for images, they are usually separated by the domain used for data embedding. The easiest approach (first introduced in [KM92] and [T^+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 which embed data in the frequency domain [CW01] or in the wavelet domain [MU01]. A recent overview on techniques applied in information hiding is given in [Z^+07].

3 ILLUSTRATION WATERMARKING FOR 2D DIGITAL IMAGES

The main idea to realize our concept of interactive images is to store an image 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.

In our approach, we focus on RGB color images with 24 bits (8 per color channel) per pixel. Due to the requirements of a high capacity, transparency, blind detection, and region-based encoding, we choose an approach based on embedding data in the pixel's least significant bits (LSB). A high capacity is expected because 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. Size of the bit stream and data type are stored in a seed point at a prominent location within the watermarked region.

3.1 Adaptive Data Embedding

The basic LSB approach performs watermarking independently from the content of the cover medium. But the capacity is not constant among pixels; it depends on the features of the cover image. To maximize the capacity, the number of bits replaced in pixels should be adapted to the image content.

Hence, we introduce a so-called *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, we separate image data used to build the capacity map 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. Hence, the 4 most significant bits maintain the original values and are suited for generating the capacity map. Due to this condition, a maximum capacity of 12 bits per pixel is given, 4 in every color component. In the following, it is described how texture and color can be analyzed to build an image's capacity map.

The texture of an image has a strong effect on the data hiding capacity. The more homogeneous the texture, the easier content modifications are visible. A homogeneity measure is required to estimate the capacity of an image region.

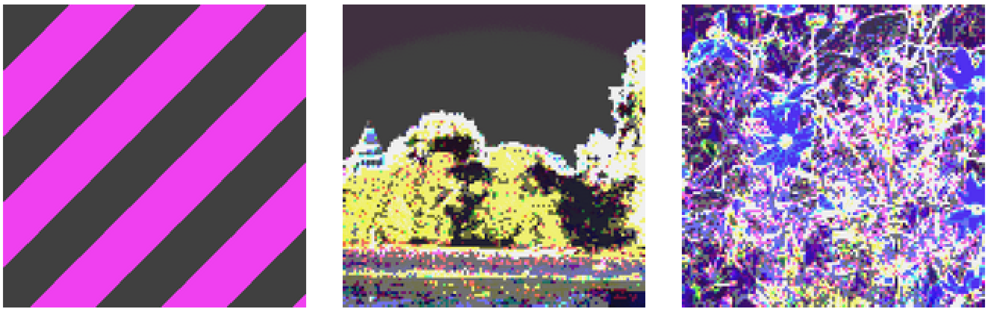
Droogenbroeck and Delvaux [VD02] introduced an approach for gray-scale images which computes the capacity from the entropy of small image blocks. Since in our approach only the 4 most significant bits of color components can be used, we adapted the algorithm to meet this requirement as described in the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

For every small image block, 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, $i = 1, ..., L$ which can occur, entropy is defined as

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

The entropy value is compared to thresholds to define the capacity of the color component for all pixels of the image block. If the entropy is larger than half of the maximum entropy, a capacity of 4 bits is assigned. Otherwise, the entropy value is compared to a quarter of the

(a) Green Pattern (b) Landscape (c) Flowers



(d) Green Pattern (e) Landscape (f) Flowers

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.

maximum entropy to assign either 3 to 2 bits as the capacity. The thresholds were chosen 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 inhomogeneity of the image texture.

In homogeneous regions, the capacity computed from the image 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. A distance between two colors in RGB cannot be mapped to the 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. Well known examples are CIELAB and CIELUV [Sto03]. These color spaces represent perceived distances with Euclidean distance values.

The capacity of a color 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, the 4 most significant bits of a color can be used to compute its capacity. The same capacity must be assigned to all colors sharing the 4 most significant bits, which is considered as a color class. Thus, to find the capacity of a color class, the smallest capacity of all colors within that class is chosen.

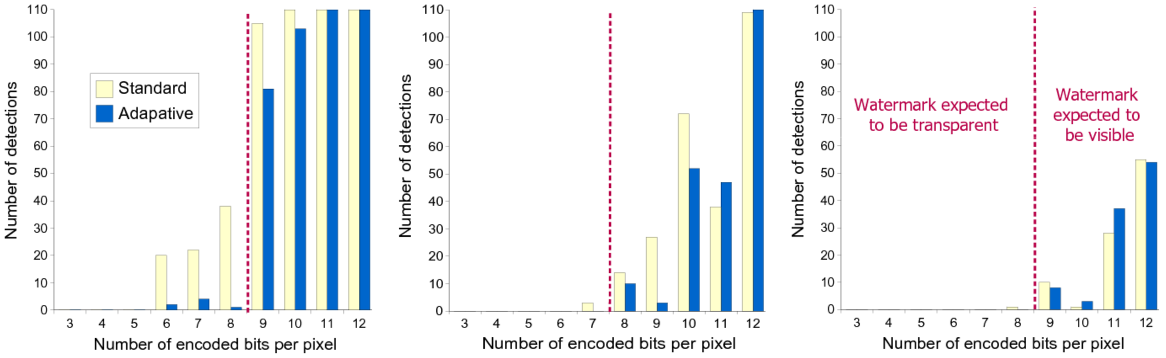
To combine texture and color analysis in one algorithm, both capacity values are computed for every pixel' color channels and the higher value is chosen as the capacity.

3.2 Evaluation

To evaluate the results of the adaptive technique introduced in the last section, a user study was conducted with two main intentions. One was to determine the reliability of computed capacities by comparing them to evaluated capacities. Besides, it was 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, amount of encoded data and encoding technique. Bit streams of different sizes were encoded to determine the maximum capacity for a constant image quality.



(a) Green Pattern (b) Landscape (c) Flowers

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 threshold where the watermark was expected to become visible.

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

One equally sized and shaped region was watermarked in every image. Bit streams of different size were encoded in regions with comparable computed capacity. To compare standard and adaptive LSB technique, identical locations were chosen for watermarking if the same amount of data was encoded in one image type.

112 volunteers participated in the study. The study was conducted on a standard PC with a high definition TFT flat screen. All images were presented for a maximum of 10 seconds in a random order, one at a time, with the three images 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

One intention of the study was to evaluate the computed capacities from our adaptive approach based on the following assumption: For encoded data volumes smaller than the capacity, the watermark was expected to be transparent and visible otherwise. Results of the study (Figure 2) show a coherence between expected visibility and the number of detections (blue bars). For the images *Landscape* and *Flowers*, none of the participants detected regions which were 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 approach to our adaptive approach, a significant statistical difference in the number of region detections was shown for the images types *GreenPattern* and *Landscape*. Watermarks coded with the standard LSB method (yellow bars) were more often detected (*GreenPattern*: 520, *Landscape*: 263) than those coded with the new adaptive method (*GreenPattern*: 415, *Landscape*: 222).

The adaptive approach has two advantages compared to the standard approach. First, it gives an estimation of the capacity of an image region. In an application scenario, the amount of descriptive information that can be embedded in an image can be limited to ensure a transparent watermark. 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 data volume of 7 to 8 bits per pixel that can be encoded.

4 INTERACTION WITH EMBEDDED DATA

While in the last Section 3, technical aspects about metadata storage have been discussed, the user interaction with the metadata is an important topic. First, the user must be given hints about the - per default hidden - metadata which is 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.

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

Considering the first aspect, visual cues are needed to communicate the existence and the properties of auxiliary data. The metadata should be easily detectable, context occlusion avoided. Also, a clear correlation between cue and augmented region is necessary. As important properties of the data, 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 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. In this section, we introduce two approaches to support the viewer in the exploration of an interactive image, the Informative Cursor and the Meta-Previewer.

4.1 The Informative Cursor

The prototype of an Informative Cursor is the well known mouse cursor which can change its style to indicate a certain action that can be performed. Informative cursors are an extension of this concept. Beside signalizing the current position of the input device, it utilizes arrows to give user hints about regions with associated metadata.

Other graphical attributes encode further properties of the metadata, as shown in Figure 3. Two regions have been annotated with additional data. The first region, showing daisy blooms, 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 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 highlighting its 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 that of a standard cursor.

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.

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$^+$93]) because the Meta Previewer is placed in front of the image to display additional information associated to the covered image region. The Meta-Previewer is a small rectangular shape. It 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 to the distorted version of the image, a frame provides further information about the metadata associated with the focused region.

In Figure 4, two examples for the Meta-Previewer are given. In the first example (Figure 4a), the mouse cursor is placed over the dandelion leaf in the lower image region. Hence, the Meta-Previewer is placed there as well. Within the Meta-Previewer, 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 (*T*ext, *I*mage, *A* for combinations). For the focused region in Figure 4a, text is provided as metadata which correlates to the user's interest. From Figure 4a to Figure 4b, the mouse cursor, and thus the Meta-Previewer is moved to a second region. Here, text and image are 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*.

(a) (b) (c)

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

Providing a content-based exploration of images, additional information such as texts and images can augment specific objects in the image. In Figure 5a, employing the *Illustration Watermarking* technique 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 opaque liver can still 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 how 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 which includes translations of basic terms, the user can switch between object descriptions in different languages. The examples were generated with the software tool *Smage* which can be downloaded at http://www.smage.de.

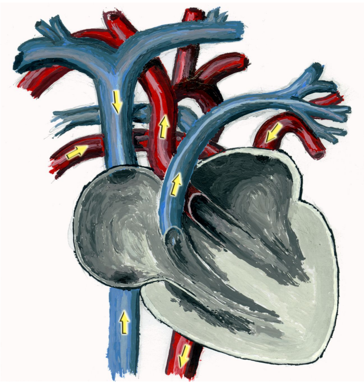


Figure 6: Illustration of the heart with arrows that change their positions.

So far we have shown examples with static metadata which could either be displayed or not. We also see application scenarios in which metadata content not only appears on demand but in which it also changes over time. All the information required for such application scenarios is stored as *Illustration Watermarks*. In addition to the components to be displayed (display data), the metadata also contains information about how the content is to be animated and how it can be affected by the user (animation data). An example for such an 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 advantages 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 computer game. The animated components are the apples and the arrow. The archer can be controlled by the user. The basic goal of the game is to hit each apple with the arrow before the apple comes off. The user can adjust the direction of the bow and trigger the shots.

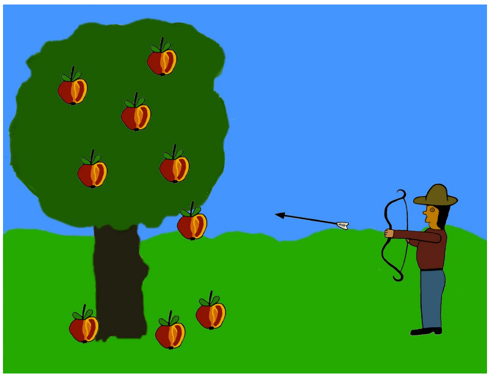


Figure 7: Image with a simple computer game encoded.

6 CONCLUSION

The concept of region-based *Illustration Watermarking* for 2D raster images has been proposed in this work. Two main challenges were adressed. First, a technique to store and retrieve descriptive information in an image has been described and evaluated. A minimum of 5 bits per pixel can be embedded without decreasing the image 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, for example JPEG2000. Since techniques to store metadata content-based have already been developed for these format [Leh07, K^{+}06], we will extend our *Illustration Watermarking* framework to embed metadata in raster images as well as in lossy formats based on the wavelet transform.

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^{+}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.

[C^{+}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^{+}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)*, page 201Ű208. 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^{+}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.

[RD04] R. Reinhardt and S. Dowd. *Macromedia Flash MX 2004 Bible*. John Wiley & Sons, Ltd., 2004.

[S$^+$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$^+$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$^+$07] Dong Zheng et al. A survey of rst invariant image watermarking algorithms. *ACM Comput. Surv.*, 39(2):5, 2007.