Adaptation of document images to display constraints

Kathrin Berkner[*] and Berna Erol
Ricoh Innovations, Inc. 2882 Sand Hill Rd, Suite 115, Menlo Park, CA, 94025

ABSTRACT

The variety of displays used to browse and view images has created a need to adapt an image representation to constraints given by the viewing environment. In this paper various methods of adaptation to a small display size are introduced with focus on adaptation of document images.

Compared to photographic images, document images pose an even greater challenge to represent on small size displays. If a typical down-sampling of image data is performed, we not only loose some high-resolution data, but also semantic information, such as readability, recognizability, and distinguishability of features.

We explore various ways of controlling document information such as readable text or distinguishable layout features in different visualizations applying specific content-dependent scaling methods. Readability is preserved in "SmartNails" via automatic content-dependent cropping, scaling and pasting. Content-dependent iconification is proposed to provide distinguishability between layout features of document images. In the case of multi-page document content a rendering in form of a video clip is proposed that performs content-dependent navigation through the image data given display size and time constraints.

Keywords: Thumbnails, display adaptation, automated scaling and cropping, task-dependency

1. INTRODUCTION

With the increasing variety of devices used in everyday life the need for adaptation of images to devices has gained a lot of interest in recent years. Especially on mobile devices displays are often very limited in size and pose challenges to traditional image viewers. When a traditional scaling of images is performed to target a small display, details in images are often lost. This is a particularly big loss in document images where text is present representing semantic information when text is readable and its layout is recognizable. If text becomes unreadable or layout unrecognizable due to too much scaling a lot of the semantic information contained in a text document page image is lost.

In recent years, a variety of resizing algorithms have been proposed, mostly for photographic images. In this paper, we review those resizing techniques and look into how they treat resolution properties of specific image features under scaling in Section 2. Then we introduce three techniques that focus strongly on the aspect of content-dependent scaling, i.e. scaling of image regions depending on the visual and semantic content of that region. One method, SmartNails (Section 3), focuses on presenting readable text, the second method, Dynamic Document Icons (Section 4), adjusts the size of the final image such that layout units, that have been selected by a user or by some algorithm, are visually separable, and Multimedia Thumbnails (Section 5) provides a solution to the problem of presenting a multi-page document on a small display given a certain time limit for browsing through pages.

When exploring the use of different resizing techniques in tasks such as searching, browsing, or document understanding, it has become apparent that resizing techniques should depend on users' tasks. The user needs for document visualizations could be categorized into the following four groups.

1. The genre or overview look of a document is enough for the user to make a decision based on seeing the visualization.
2. The user needs to read or view details on a certain page in order to make a decision.
3. The user needs to read or view details on a specific page that depends on a recent activity, e.g. a search term.

[*] berkner@rii.ricoh.com; phone 1 650 496-5714; fax 1 650 854-8740; rii.ricoh.com

4. The user needs to see differences between documents in order to make a decision.

These user needs are often neglected in the design of image resizing algorithms. The task dependency of resizing algorithms and their relationships with user needs are discussed in Section 6. The paper concludes in Section 7.

2. IMAGES ON SMALL-SIZE DISPLAY

Paginated documents are traditionally designed for being read on paper. A high-resolution scan of a document contains a lot of the information that a printed paper copy would contain. Text is readable, image content is recognizable, and the layout or page format has not been changed through scanning. The only change in appearance might be the loss of color if a grayscale scanner has been used. When viewing the digital version on a large-size and high resolution display not much of the original document information is lost except the ability to flip through the document in the same effortless way as through the paper copy. On the other hand, there is a big loss of information when digital documents are viewed and navigated on devices with small display areas; if a whole document page is displayed, the text is usually not readable, if text is in readable form, the layout information is missing and it becomes difficult to recognize and navigate through the content. This kind of information loss needs to be considered when creating images for small displays.

Various resizing techniques for adapting high-resolution document images to small displays exist. In general, image resizing can be done via scaling, cropping, or a combination of both. In traditional thumbnail-creation only scaling is applied. A scaling parameter s is computed from the ratio of the original image dimensions (width and height) and the dimensions of the target image (thumbnail). Then scaling is performed by applying an anti-aliasing filter to the image and then down-sampling the image by keeping every $1/s$-th sample. Characteristics of the anti-aliasing and down-sampling filters are described in the Fourier-domain. That means global frequency characteristics of the image signal determine the resizing procedure. As a consequence, no locally varying or semantic information is controlled via traditional scaling. Specific image features of local structures or semantic information in certain regions may or may not be lost. This missing explicit control over information loss has a big impact on the usefulness of thumbnailing of document. If text becomes not readable anymore after resizing, a lot of semantic information is lost.

Besides traditional down-sampling, a few other methods have been proposed to fit image content to small-size displays that will be discussed in the following. One of the first technologies to address the problem of repurposing document images to small displays are Enhanced Thumbnails[1] and Paper-to-PDA[2]. Enhanced Thumbnails technology creates an enhanced version of a traditional thumbnail by first down-sampling the image information, lowering the contrast of the thumbnail, and then pasting augmented keywords onto the thumbnail image, maintaining the original location of the words on the page as best as possible.

A change of layout focusing on the text regions of a scanned document page was considered in the Paper to PDA technology[2]. In a first step the digital document image was segmented into regions containing individual words. Then given a display width, the words segments were repositioned to produce re-flowed text.

More recently, repurposing of photographic images to fit small-size displays has become a research topic of interest. The goal of creating thumbnails that have similar blur and noise characteristics as the full-resolution image is pursued in the Honest Image Thumbnails technology[3]. Blur and noise extracted from the original image is added to the thumbnail to keep the same appearance and is applied to photographic images. For document images, that might have been created via a printer driver from a text-editing tool or from a document scanner, it is not clear whether or not significant blur can be found in the original image. For images taken by a handheld camera, in contrary, blur created by motion or out of focus is an issue. It still needs to be investigated, however, whether and in which usage scenario, introducing such a distortion into document thumbnails is useful.

An automatic cropping of certain regions in photographic image has been proposed in various publications. The technology Seam Carving[4] minimizes energy functions along certain paths of small gradient to find final horizontal and vertical "seams" are determined that divide the original image into four parts according to the image content.

Penalties for information loss were introduced in the Video Retargeting[5] technology. These penalties are computed from loss of pixel values through cropping, penalty for scaling with a factor smaller than one, and for changing the aspect ratio.

An alternative automatic cropping method was used in the Photobrowsing technique[7] based on saliency maps. A minimal perceptible size for extracted objects was defined. That size, however, was supposed to be set by a user (author or publisher). The size was not directly computed from the object itself.

Besides the traditional thumbnailing, all other reviewed techniques perform some kind of processing to solve the problem of "I don't see anything in this thumbnail image". They provide, however, no or only little control over how to resize such that certain image features and semantic information in the document are still perceptible by the viewer.

Some of the techniques perform only cropping with no additional scaling. In that case some context information may be lost, but full-resolution properties of objects in the cropped image are maintained. Others[2] perform scaling, but do not consider the resolution properties contained in the area under scaling. For e.g. in Video Retargeting, a penalty function dependent on a preset scaling factor is used and is not dependent on image content. In Paper-to-PDA a global scaling factor was chosen that worked for a 12pt font type. Similarly, a fixed minimal size was set in Photobrowsing for a certain class of objects. Our work on SmartNails[8,9], Dynamic Document Icons[11], and Multimedia Thumbnails[12,13] addresses that question of how to assign a proper size to image objects by including content-dependent resolution information into a resizing algorithm and therefore adding more control over the perceptibility of complex image information into the algorithm.

In the following, we will review the resizing technologies SmartNails, Dynamic Document Icons, and Multimedia Thumbnails and focus on incorporation of content-dependent resolution information into the document retargeting algorithms.

3. SMARTNAILS

SmartNail technology[9] performs content-dependent scaling, cropping, and repositioning of document image objects. In the first step, a document image needs to be segmented into semantic units, such as text paragraphs, section headings, background image, figures, figure captions, footnotes, etc. Then an optimization step follows that has the goal to compute best scaling, cropping, and repositioning parameters for each image object given a certain cost function. In order to achieve such goal, the segmentation method has to extract some information related to scaling, cropping, and repositioning operations. Since document images typically contain text and image content, two different segmentation techniques are applied, that focus on the different characteristics of text and images, followed by a step of merging the two segmentation results into one. The segmentation technique for text regions applies layout analysis as it is used as a pre-processing step for OCR to find bounding boxes of paragraphs, lines, words, and characters. Depending on the height distribution of character bounding boxes in a given text region, a minimal scaling factor is assigned to the region following the criterion that after scaling a text box should not be less than 7 pixels high. This threshold of 7 pixels is set such that the final resized text can be readable on a 100dpi cinema display screen. In this way, the minimal scaling factor for a text object is chosen depending on its content with the goal of preserving readability.

The segmentation technique for image regions performs an analysis of a multi-resolution bit distribution. Such a distribution consists of bits allocated to code different regions of an image at various resolutions. One way of obtaining such a bit-distribution is to extract information from a JPEG 2000 (J2K) encoded file. A J2K encoder wraps individually coded wavelet domain blocks of coefficients into units called packets. These packets get a header attached describing the length of the coded data following as well as some other information such as code-block inclusion and zero-bit-plane information. The length of the coded data corresponds to the number of bits the coder spent on the wavelet domain block. Since the wavelet domain is a multi-resolution domain, blocks at different locations in the domain correspond to bit numbers spent at various resolutions. For more details on the multi-resolution bit distribution (MRBD) derived from J2K headers we refer to[8]. Each location in the MRBD gets assigned a preferred resolution by choosing the resolution that contributes the most amounts of bits to the location. Once that block-based resolution assignment is done, a connected-component analysis is performed to group neighboring blocks into connected components to form a *resolution-sensitive saliency map*. As a result, each component has a resolution attributed attached to it. This resolution attribute r is used to determine an appropriate scaling factor s_I for the image segment associated with the wavelet-domain via the relationship $s_I = 2^{-r+1}$. This scaling factor is not the one used in the final scaling of the segment, but serves as a guideline of how to best scale the segment.

As a result, each image object, extracted from the resolution-sensitive saliency map, has an appropriate scaling factor associated with it. The resolution-sensitive saliency map is also used to assign an information value to each object. The information value is computed as the number of bits necessary to code the object at its preferred resolution.

In an optimization step, segments are selected that fill the available SmartNail area after possible scaling, cropping and repositioning as best as possible according to the minimization of a cost function. During the optimization, scaling with a factor that deviates from the appropriate scaling factor is allowed, but is penalized according to the amount of deviation. Similarly, change of relative positions between segments deviating from the original relative positions is penalized.

For a text object, cropping is an allowed operation in the optimization step, where cropping means removing words from the end of the text unit. Penalty for that cropping is the subtraction of the sum of the information values of the removed words. Details on the optimization procedure including cost function and greedy algorithm are contained in[9].

In Fig. 1 several examples of applying traditional thumbnailing and the SmartNail technology to images to resize them to fit a small area (130x160 pixels) are shown. In examples (a), (b), and (c), the text-focused layout analysis did not detect all the text because some of the text was in a rather graphical font or in a shaded gray color instead of black text of a common text font. In these cases, the results from the segmentation technique targeting image regions applied to the problematic regions were merged with the text-based segmentation results for the remaining regions. In example (d) the layout analysis failed completely since it was not designed to work on light text on dark background. In this case, the segmentation technique targeting image regions was applied successfully. In (e) the advantage of considering a bit distribution over image regions becomes transparent. In order to code the grey figure to the left some bits are necessary for the top part, but almost none are needed for the remaining constant grey rectangle. Our MRBD-based segmentation algorithm detected that most information was concentrated in the top of that rectangle and cropped that part out. That way there was space in the final SmartNail underneath the gray figure to place text, instead of placing it to the left of the figure. Example (f) shows a case where the SmartNail algorithm has not done as nice as a job as in the other ones. Layout analysis did not give any good segmentation of the text since the text was too large, text segments too few and too scattered across the page. Only the text on the top "Get Wild About" was detected. The remaining three included segments were found via the image segmentation methods. The block-based nature of that segmentation method is visible in the segment that includes the text "A Guide for …." The text is not being segmented perfectly. Some of the big letters belonging to the word "READING!" are still included. The word "READING!" itself was not selected by the algorithm.

Adaptation to three different display sizes, 130x160, 80x100, and 400x200 is shown in Fig. 2. The smallest SmartNail shows how text is re-flown in order to optimize the use of the thumbnail area. The 400x200 SmartNail was designed for displays found, e.g., on a copier control panel. In that case, the changes of layout were bigger than in the case of the aspect ratio of original image and SmartNail being the same.

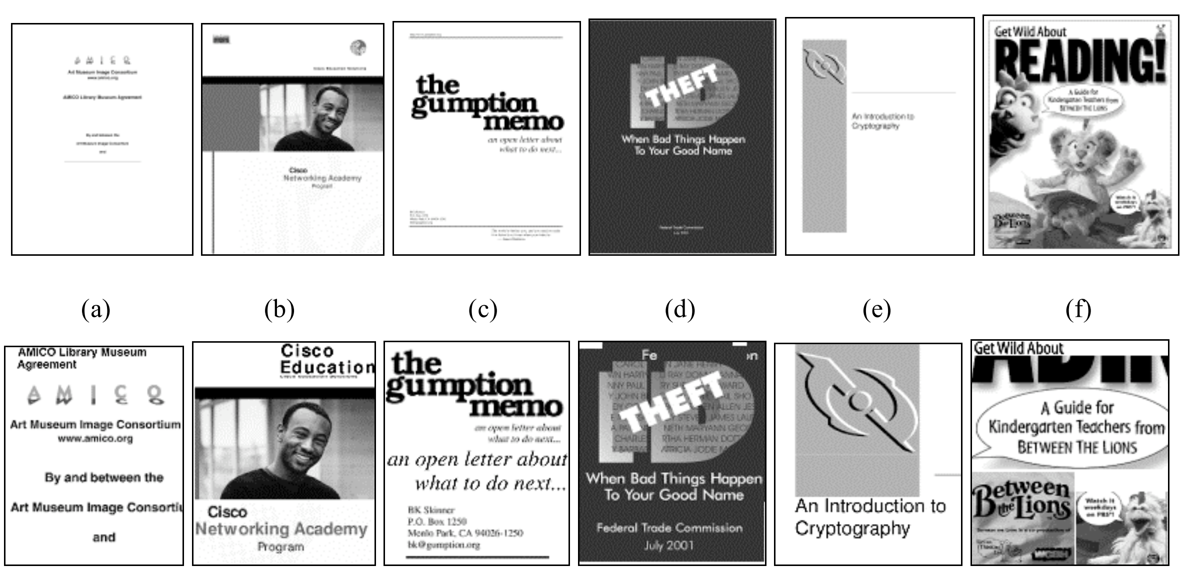

Fig 1. Image resizing to fit an area of 130x160 pixels with traditional thumbnailing (top) and SmartNail technology (bottom).

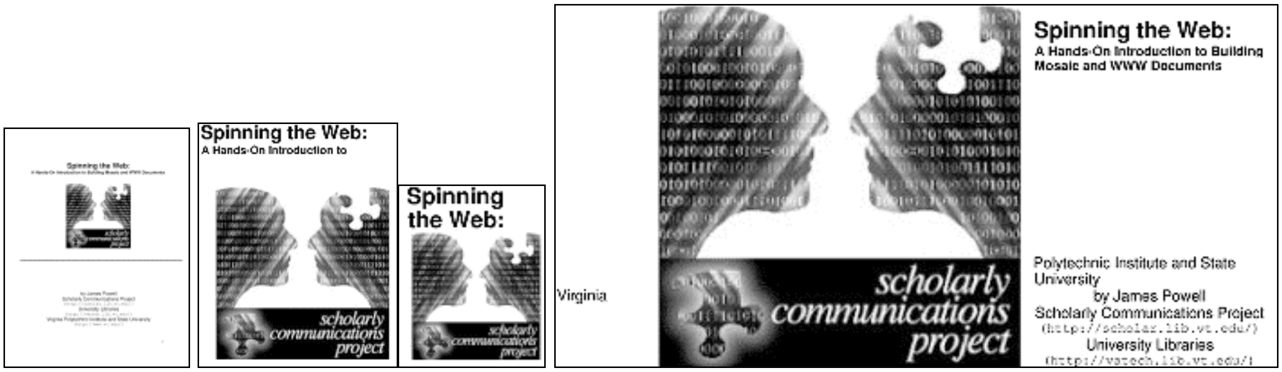

Fig. 2. Demonstration of SmartNail technology for varying display sizes. From left to right: traditional thumbnail for 130x160 pixels, Smartnails for area size 130x160, 80x100, and 400x200 pixels.

4. DYNAMIC DOCUMENT ICONS

When investigating the use of SmartNails in a document retrieval scenario the users commented that it might be distracting or confusing to the user if the layout of document elements in a SmartNail is very different from the layout in the original document. As a result, we posed the question whether it was possible to develop a resizing algorithm that focused solely on the preservation of certain layout features and did not try to present text at a readable resolution.

Our solution to the problem of layout-preserving resizing is based on the strategy of switching from a scaling of image intensity data of the original image to an iconic representation of layout units. Since readability of text is neglected the information loss when going from scaled images to iconic representations is not considered to be very high. Another advantage of using iconic representations is that certain information losses such as loss of separation of two layout units due to too much scaling can be easier controlled. An important feature of the iconic representation is that its size depends on the layout content intended to be visualized. If a lot of small layout units should be separately visible with separating white-space after resizing, a larger geometric size of the final image is needed. If only coarse units should be visible, a smaller geometric size may be allowed. The final outcome of this layout-preserving resizing technology is called a “Dynamic Document Icon” (Dydocon), where “Dynamic” stands for the content-adaptive geometric size of the icon.

A Dydocon is composited out of a set of iconic elements that may have a simple geometric shape, such as a rectangle, and each of the elements has a *minimal geometric size* (MGS) assigned to it. For example, a text unit may be represented by a rectangle of constant color with its minimal geometric size being a width and height of 7 pixels for display on a standard LCD laptop or desktop screen. The background of a document image may be represented by a rectangle of certain aspect ratio with a minimal geometric size similar to that of a text unit. The fill-color of the rectangles may depend on the document content. A unit of black text in bold font style may be filled with a darker gray than a unit of black text in regular font style. The rectangle representing the background may be filled with the average color of the background in the original document image. A figure may be represented by a rectangle of the same aspect ratio with a fill-color equal to the dominating color in the figure in the original document image or filled with a downsampled version of the original figure. Given a decision is made of what layout units of a page to include in the final Dydocon, those units are selected, its associated iconic elements composited according to the original page layout and scaled such that in the final composition of iconic elements no element is scaled to a size smaller than its minimal geometric size.

Examples of document images and their associated Dydocons are shown in Fig. 3. Example (a) is the example of a one-column document with an image and a title. The background was represented by a white rectangle, the title unit by a black rectangle, the complete one column with a gray rectangle, and the figure-image was copied and scaled. The limiting MGS of the elements is the one associated with the title rectangle. In (b) and (c) the limiting MGS is the one of the separating white-space between the two columns, but since (c) has three column elements the Dydocon is larger than that in (b). In (d) the title element has the limiting MGS. But compared to (a) the three columns require a larger size than the one column.

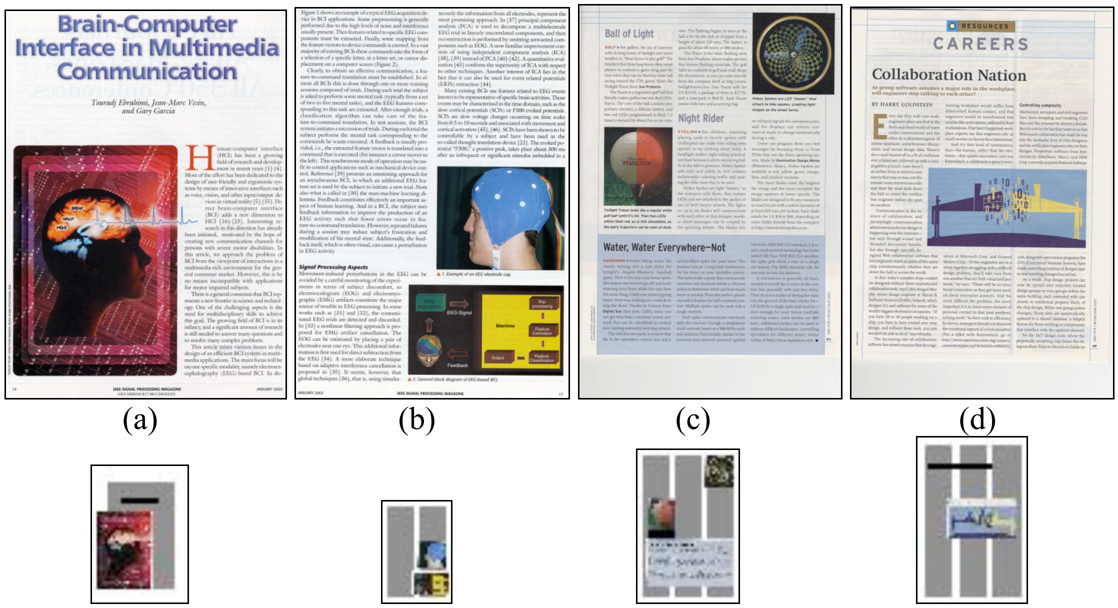

Fig. 3 Dydocons for various document pages.

The scaling may not only depend on each included element's minimal geometric size, but also the specific composition. For example, if a light red rectangle representing a figure is placed close to a gray rectangle representing a text unit with some separating white space in-between, the contrast between those three elements may become invisible even when scaling to a size greater than the largest MGS of all the element since it is well understood that color contrast changes with decreased pixel resolution as modeled, e.g., in S-CIELAB[10].

Since layout features may be the same or very similar for a variety of document images, Dydocons can also be used to represent a group of document images. A solution to how to adapt a Dydocon representation of a group of images to a given fixed display width is presented in [11].

5. MULTIMEDIA THUMBNAILS

All the techniques described so far deal with resizing of one single image. Electronic documents, however, often consist of multiple pages, i.e. multiple document images. In this case not only visualization of the information on one page, but of information on multiple pages and relationships between pages becomes important. A Multimedia Thumbnail[12,13] (MMNail) is a short video clip presenting a guided tour through a document. An MMNail visualization uses both visual and audio channels as well as time dimension to communicate document information by automatically zooming into important document parts, such as abstract, title, and figures, and converting some document parts to audio with synthesized speech, such as keywords and figure captions, as illustrated in Fig. 4.

The MMNail algorithm consists of three parts, an analyzer, an optimizer, and a synthesizer. In the analyzer important document regions are detected and information, resolution, and time attributes assigned. The optimizer takes this information in addition to application and display constraints and selects an optimal subset of regions and associated visualization techniques to be included in the final MMNail. The synthesizer then takes all that information and composes the final video clip (including audio-synthesis).

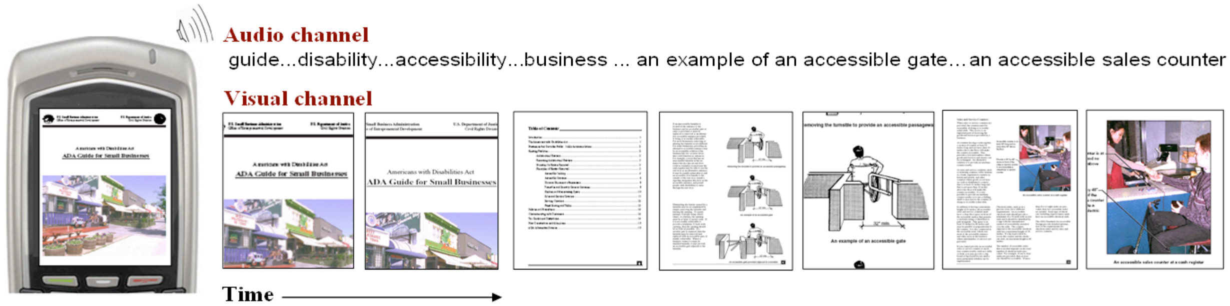

Fig 4. An example of a Multimedia Thumbnail containing visual and audio content. The visual content includes animation operations such as zooming into a title or figure caption. The audio channel plays keywords extracted from the document.

Resolution sensitivity of the algorithm is introduced in the analyzer and optimizer. The analyzer computes for each image object *I* an appropriate resolution *r(I)* in the same way as it was done in the SmartNail technology. These resolution attributes then enter into the calculations of time attributes. Each image object obtains a time attribute *t(I)* that determines how much time is needed during viewing to perceive the image information. In order to determine *t(I)* first traditional thumbnailing is performed to fit the entire document image to the display. Then a zoom factor *Z(I)* is determined as the factor that is necessary to scale the object to its appropriate resolution. In particular for text objects, perceiving the information may mean that a panning operation during rendering may be needed. Such panning time depends on the geometric dimensions of the object and the target display size. In addition to zoom and pan time, object content, such as numbers of characters *N(I)* in the object, enter into the time attribute as well. As a result, the overall time attribute for an object is defined as

$$t(I) = \left\{ \begin{array}{ll} C \times N(I), & Z(I) = 1 \\ C \times N(I), & Z(i) > 1 \end{array} \right\},$$

where *C* is a constant measuring the average time the speech synthesizer needs to synthesize a character.

For image objects, the parameter *N(I)* can be determined as a comprehension time depending on the number of bits in the image. For images containing graphics, *N(I)* should depend on a mixture of number of bits necessary to code the image and the numbers of readable characters in the graphic.

Multimedia Thumbnails is not the only technology that includes a time dimension into a resizing algorithm. The Photobrowsing technique[7] also considers time and computes an optimal path through a set of objects that can be displayed in the given time. Time attributes are assigned differently to text, face and general image regions. Whereas for text regions the time attribute depends on the number of words, in a face region it depends on the number of faces, positions, poses, etc. Photobrowsing, however, does not consider time for zoom and pan operations.

6. TASK-DEPENDENCY IN IMAGE RESIZING

When we started to investigate how people would use various small-size document visualization techniques it became clear that there is not a single visualization that fits all needs. In every usage scenario the visualized image was a stimulus for making a decision, e.g. clicking on an image or refining a search query. The user needs for document visualizations could be categorized into the following four groups.

1. The genre or overview look of a document is enough for the user to make a decision based on seeing the visualization.
2. The user needs to read or view details on a certain page in order to make a decision.
3. The user needs to read or view details on a specific page that depends on a recent activity, e.g. a search term.
4. The user needs to see differences between documents in order to make a decision.

SmartNails address the needs stated in (2) and (3) and did not work very well when an overview (need (1)) was required. In general, computers can not always predict what the user's specific need is. Therefore, the user should have the opportunity to choose between different visualizations. In an internal lab experiment we added SmartNails as pop-ups to conventionally created thumbnails as shown in Fig. 5.

Dynamic Document Icons initially started out trying to satisfy user need (4). The solution, however, is not quite satisfying yet. Visualizing images in a way such that differences between displayed images are perceivable is still an open research problem. Differences may be of semantic nature and depend on a specific application or they may be purely image-based and need to be measured in terms of object separation, contrast, color differences etc.

Multimedia Thumbnails address user needs (1) and (3) and include multi-page documents. In the design phase of the algorithm the two specific usage scenarios of document understanding and document browsing were investigated. The findings from user studies resulted in different parameter settings for the MMNail algorithm for the two scenarios[13]. For example, image content in figures was important in the document browsing task, whereas figure captions were more important in the document understanding task.

Task and specific viewing environment properties may also influence the geometric size constraints given to the resizing algorithm. As we saw in SmartNails, a selection of elements had to be made in order to fit the content to fixed target size constraints. The selection and possible change in layout was not the right information to convey when the user needed an overview. That means it would have been more appropriate for the overview task to relax the tight size constraints and compute SmartNails allowing flexible width and height of the final image and impose tighter constraint on layout changes.

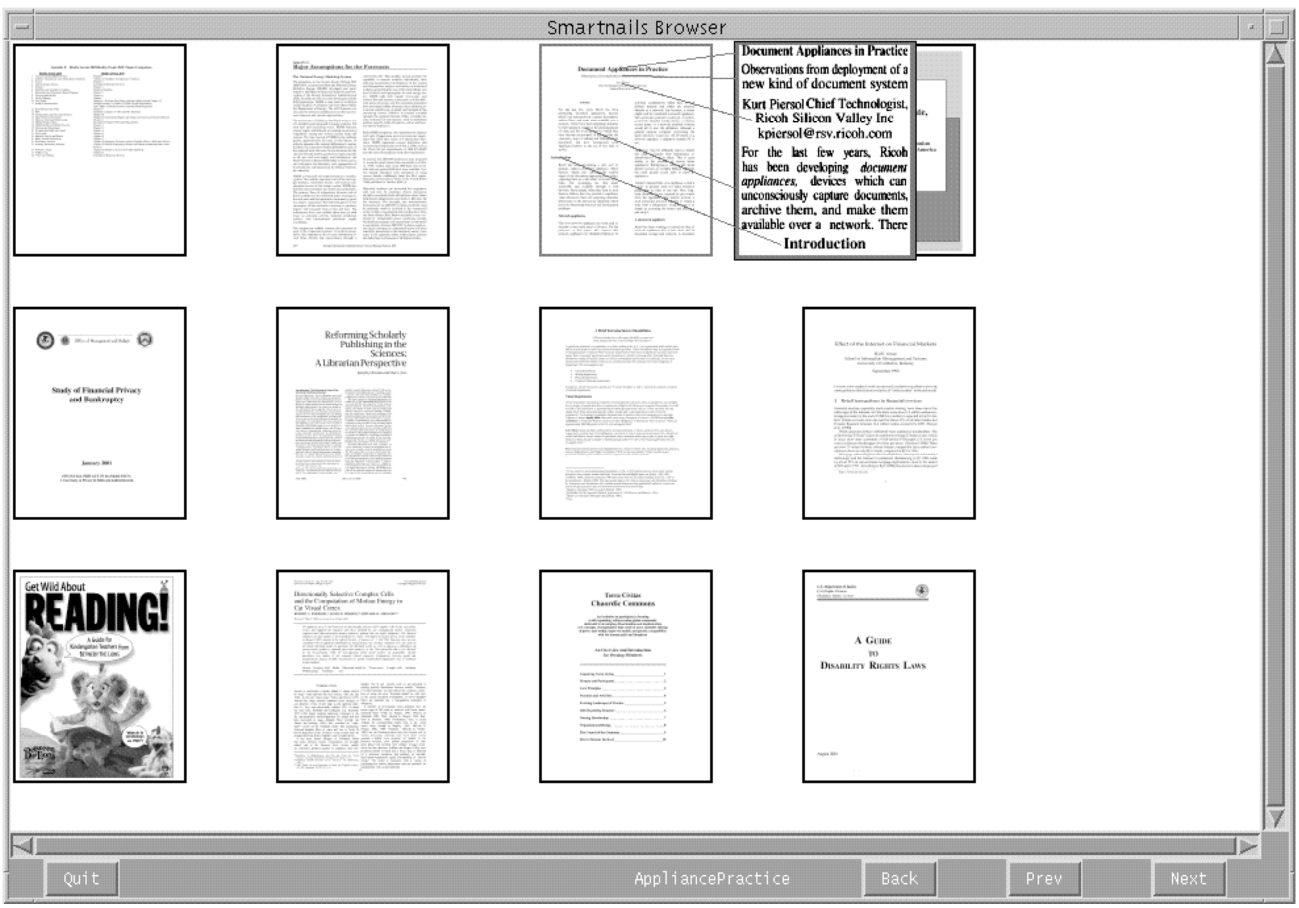

Fig 5. SmartNails as pop-ups to traditional thumbnails.

With the Dydocon technology we focused on making sure certain layout units in the image where recognizable, resulting in a flexible size for the visualization. When fixing the size only a selection of layout units can be visualized. Such a relationship between task and the geometric size of the visualization has not been considered in all the resizing techniques reviewed in Section 2, where either a fixed size was considered or a flexible size was allowed and the final size determine by a cropping algorithm that calculated object sizes. In Table 1 we summarize the dependencies on content and tasks of the various resizing techniques reviewed in this paper.

7. CONCLUSIONS

In this paper we reviewed various resizing methods for images for viewing on small displays. Although automatic cropping could be found in a variety of methods, content-dependent scaling was not applied widely. We described the content-dependent scaling aspects of three resizing methods, SmartNails, Dynamic Document Icons, and Multimedia Thumbnails. All of them are applied to document images that may contain text and image content. Any one of those methods cannot be successful in a wide range of different tasks such as searching, browsing, or understanding, since the small display area restricts too much information. Therefore, it was important to understand what kind of document image information was controlled by the resizing algorithm, readable text, distinguishable layout, or multi-page overview. Only once these characteristics are understood and matched to certain tasks a single method or a combination of methods can be chosen to assist the user in fulfilling the task. Such task dependency in image resizing or in general image understanding algorithms is still under-explored and poses a lot of challenges for future work.

Table 1. Overview of dependencies on content and task considered in various resizing techniques.

	Content-dependent cropping	Content-independent scaling	Content-dependent scaling	Task dependency
Traditional Thumbnailing via down-sampling		x		
Enhanced Thumbnails		x		
Paper-to-PDA		x		
Honest Image Thumbnails		x		
Seam Carving	x	x		
Video Retargeting	x	x		
Photobrowsing	x		x (not automatic)	
SmartNails	x		x (automatic)	x
Dydocons			x (automatic)	
Multimedia Thumbnails	x		x (automatic)	x

REFERENCES

1. A. Woodruff, A. Faulring, R. Rosenholz, J. Morrison, P. Pirol, “Using Thumbnails to Search the Web,” in Proc. of SIGCHI’01, 2001.
2. T. Breuel, W. Janssen, K. Popat, H. Baird, “Paper to PDA,” Proceedings of 16th International Conference on Pattern Recognition (ICPR’02), vol. 4, pp. 476-479, Quebec City, Canada, 2002.
3. R. Samadani et al., “Representative image thumbnails for good browsing,” Proceedings of ICIP2007, pp. 193-196, 2007.
4. S. Avidan, A. Shamir, “Seam Carving for Content-Aware Image Resizing,” ACM Transactions on Graphics, Volume 26, Number 3, SIGGRAPH 2007.
5. F. Liu, M, Gleicher, “Video Retargeting: Automating Pan and Scan,” Proceedings of ACM International Conference on Multimedia, pp. 241-250.
6. S. Avidan, A. Shamir, “Seam Carving for Content-Aware Image Resizing.” ACM Transactions on Graphics’07, 26(3), article 10.
7. Z. Xie, H. Liu, W.-Y. Ma, H.-J. Zhang, “Browsing large Pictures Under Limited Display Sizes,” IEEE Transactions on Multimedia, vol. 8, no. 4, pp. 707-715, 2006.
8. R. Neelamani, K. Berkner, "Adaptive Representation of JPEG 2000 Images using Header-based Processing", Proceedings of ICIP2002, vol. 1, pp. 381-384, Rochester, NY, 2002.
9. K. Berkner, E. L. Schwartz, C. Marle, "SmartNails - Image and Display Dependent Thumbnails," Proceedings of SPIE, vol. 5296, pp. 53-65, San Jose, 2004.
10. G.M. Johnson and M.D. Fairchild, "A top down description of S-CIELAB and CIEDE2000," Color Research and Application, 28 425-435, 2003.
11. K. Berkner, M. Gormish, "Optimal Display Adaptation of Iconic Document Visualizations via BFOS-type Tree Pruning," Proceedings of ICIP 2006, pp. 2133-2136, Atlanta, USA, 2006.
12. B. Erol, K. Berkner, S. Joshi, J.J. Hull “Computing a Multimedia Representation for Documents Given Time and Display Constraints,” Proceedings of ICME 2006, pp. 2133-2136, 2006.
13. B. Erol, K. Berkner, S. Joshi, “Multimedia Thumbnails for Documents,” Proceedings of ACM Conference on Multimedia, pp. 231-240, Santa Barbara, CA, 2006.