Occlusion Costing for Multimedia Object Layout in a Constrained Window

Simon Widdowson
Imaging Systems Laboratory
HP Laboratories Palo Alto
HPL-2005-128
July 7, 2005*

layout, occlusion, cost function, saliency, slideshow

In this paper, we propose a novel method for applying image analysis techniques, such as saliency map generation and face detection, to the creation of compelling image layouts. The layouts are designed to maximize the use of available real estate by permitting images to partially occlude one another or extend beyond the boundaries of the window, while retaining the majority of visual interest within the photo and deliberately avoiding objectionable visual incongruities. Optimal layouts are chosen from a candidate set through the calculation of a cost function called the occlusion cost. The basic form of the occlusion cost is applied to candidate layout sets where the sizes of the images are fixed with respect to the window. The area-compensated form of the occlusion cost permits a more general solution by relaxing the fixed-size constraint, and allowing each image to scale with respect to both the frame and the other images. Finally, a number of results for laying out one or two images within a frame are presented.

* Internal Accession Date Only
Approved for External Publication

Occlusion Costing for Multimedia Object Layout in a Constrained Window

Simon Widdowson*
Hewlett Packard Company, 1501 Page Mill Road, MS 1203, Palo Alto, CA 94304

ABSTRACT

In this paper, we propose a novel method for applying image analysis techniques, such as saliency map generation and face detection, to the creation of compelling image layouts. The layouts are designed to maximize the use of available real estate by permitting images to partially occlude one another or extend beyond the boundaries of the window, while retaining the majority of visual interest within the photo and deliberately avoiding objectionable visual incongruities. Optimal layouts are chosen from a candidate set through the calculation of a cost function called the occlusion cost. The basic form of the occlusion cost is applied to candidate layout sets where the sizes of the images are fixed with respect to the window. The area-compensated form of the occlusion cost permits a more general solution by relaxing the fixed-size constraint, and allowing each image to scale with respect to both the frame and the other images. Finally, a number of results for laying out one or two images within a frame are presented.

Keywords: Layout, Occlusion, Cost Function, Saliency, Slideshow

1. INTRODUCTION

With the rapid rise in use of digital cameras and powerful viewing devices such as PCs, the need to display multimedia content in an attractive manner is becoming increasingly prevalent. For years, slideshow applications for viewing photographs were relatively straightforward. Display of single, sequential, images on the screen was functional and computationally lean. However, it was also bland and repetitive for the user. Eventually, slideshow applications began to include transitions between photographs, and quickly incorporated hundreds of animated scripts that would add minor visual interest, but eventually distract from the content itself. Finally, techniques for panning and zooming photographs on the screen have been developed[1]. In recent years, these auto-rostrum techniques have become quite advanced.

Now with the advent of high-resolution displays, a new opportunity for presenting photographs has become possible. By displaying more than one photograph on the screen at one time, a slideshow can be reduced in length, and made more visually interesting for the user. Current multi-photograph display techniques are principally based around grid[2] or tree-structure[3] layouts, with little or no cropping around the edges of the photograph. While this is sufficient for larger numbers of photographs, for only a few images the disparity between the aspect ratios of the screen and images will likely lead to significant wasted space on the screen.

The solution is to present the photographs in such a way that they at least partially overlap, both with each other and with the boundaries of the presentation window. This allows more efficient utilization of screen space, and allows the images to appear larger to the user. The difficulty arises in creating a layout on the screen such that there are no objectionable overlaps. For example, to cover a face in a portrait photograph would lead to a visual incongruity that would render the slideshow unacceptable.

The process of generating a slideshow based on these concepts involves three steps, as shown in Figure 1 below. First, a set of candidate layouts is generated. These layouts may be created manually by a graphic artist or someone skilled in the art, or may be generated automatically based on some set of parameters. These layouts may be described as a set of possible coordinates for each photo to be displayed, or as a set of constraints for those coordinates. The second step involves analyzing the photographs in order to understand the distribution of their areas of interest. Finally, the images are laid out in each of the candidate layouts, and a cost for that layout is determined. The layout which achieves the

* simon.widdowson@hp.com; phone 1 650 236-5203; fax 1 650 857-5331; hp.com

lowest cost is determined, either through an exhaustive search or some more efficient routine, and the images are laid out according to that configuration.

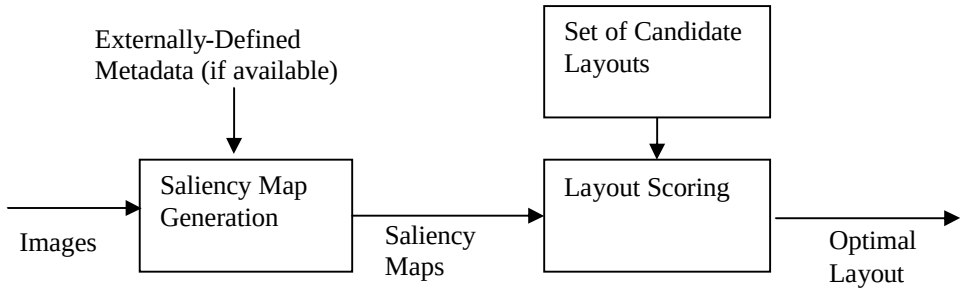

Figure 1: Process for automatically creating slideshows

This paper describes in detail the above procedure, including particular focus on the determination and characterization of the score for a given set of images in a particular layout.

2. SLIDESHOW LAYOUT PROCESS

2.1. Candidate layouts

The determination of the candidate layouts is a reasonably subjective process, with some important conditions. The size of the candidate set heavily impacts the success of the overall process. If the set of candidate layouts is too extensive, the search to find the optimal layout will take an inordinately long time. However if the set is too small, there is a risk that none of the permissible layouts complements the images to be included, and the final result will be visually objectionable. In addition, the layouts should visually appealing in some artistic sense. This means that for every set of images which is displayed at a single point in time, the layout should have some aesthetic value, for example a balanced distribution of photographs. Finally, when a number of these layouts are presented in a sequential fashion with a large collection of images, there should be minimal repetition among the layouts.

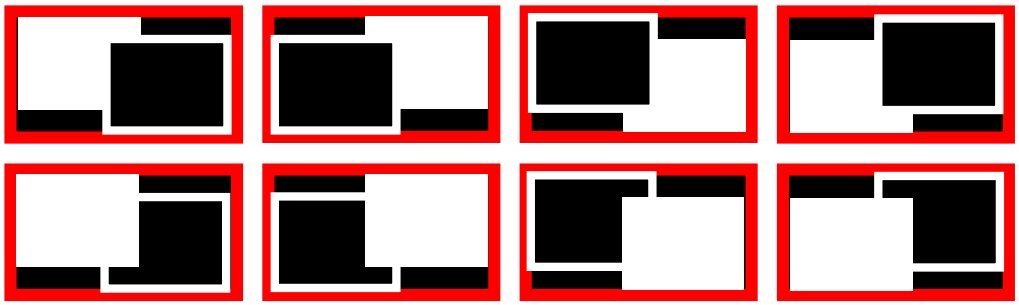

Figure 2: A trivial set of candidate layouts

A trivial set of candidate layouts is shown in Figure 2. In this case, the layouts each permit the display of two images at a time within the window. Each of these images may be placed in any of the four corners of the window, with the other image positioned diametrically opposite the other. By allowing either image to overlap the other in each of the four configurations, a total set of eight candidate layouts is created.

2.2. Image analysis

The aim of the content analysis is to determine which areas of the images are visually interesting, and which may be safely occluded. For this we rely principally on the creation of a saliency map[4], which provides a value of the conspicuity, or interest, at every point in the image. This map provides a robust indication of the areas of interest within the image. Additional algorithms may also be applied in analyzing the image, such as the detection of faces[5]. By

knowing where in the image a face is present, we can deliberately avoid any layout in which a face is occluded, and which leads to the most obvious and avoidable source of visual incongruity.

While the combination of saliency and face detection are sufficient in the vast majority of cases, there are always cases where one or both will fail, which may lead to an objectionable layout of the images within the window. The mature state of both algorithms ensures that this is a rare situation, and as both technologies continue to improve they can be trivially included in this process in order to improve overall effectiveness.

Finally, knowledge about the image can be extended by allowing the user to specify some constraints on the image, in the form of externally-defined metadata. While this can take many forms, a simple example would be the delineation of a pet in an image. Since both face detection and saliency generation might not recognize a pet as being visually important, the user may specify that a particular area of the frame is of high saliency, and should be preserved in the final layout.

2.3. Cost function

In order to determine the optimal layout from the candidate set, a scoring metric must be derived. Incorporating the concepts described above, principally that there should be no visual incongruities in the layout, we can calculate a cost for each layout. This cost will directly depend on the amount of saliency which remains visible once the images have been composited, and will thus tend to preserve visual interest in the final layout.

The cost is determined by inserting the saliency maps into the layout in order to determine which areas of each image are being occluded. By adding up the pixel values on the saliency map for the visible areas and the total area of each image, we can determine the fraction of the image saliency which remains visible in the final layout:

$$F_S = \sum_{\text{All Images}} \frac{S_V}{S_T} \quad \text{where} \quad S_V = \sum_{\text{Visible Pixels}} S(x,y) \quad \text{and} \quad S_T = \sum_{\text{Total Pixels}} S(x,y) \tag{1}$$

and where S(x,y) is the value of the saliency map at the coordinates (x,y). This states that the fraction of visible saliency depends on S_V, the saliency in a given image which remains visible, and S_T, the total saliency for that image. In each case the saliency is determined by summing the values in the saliency map. Since the goal is to maximize the visible saliency, the cost, C, of the layout, L, is simply defined as:

$$C(L) = \frac{1}{F_S}$$

These equations define the basic form of the Occlusion Cost for a given layout. This basic form allows the selection of an optimal layout from the candidate set, and creates the final slideshow.

2.4. Results for the basic form of occlusion cost

Using the set of layouts shown in Figure 2, we can use the basic form of occlusion costing to calculate the costs and determine the optimal selection for any given pair of images.

Figure 3: A pair of images to be laid out within the window shown

Figure 3 shows a pair of images which are to be arranged in the window shown. In this illustration we are using a pair of images with regular 4:3 aspect ratios, and a window with a 16:9 aspect ratio. However, it should be noted that there are no constraints on the aspect ratios, relative sizes, or other dimensions of the images or the window, other than those explicitly determined in the candidate layout set.

Figure 4: The result of a random selection (left), and the selection which minimizes occlusion cost (right)

Figure 4 shows two possible arrangements of the images within the window area. The layout on the left is the result of a random selection from the available candidate layouts. The occlusion of the subject's face creates a jarring visual incongruity. The layout on the right is the result of minimizing the occlusion cost across the layout set. In this case the incongruity has been avoided, and the subject's face is now visible. In addition, note that the gravel path towards the bottom left corner of the image, which is an area of very low interest, has also been occluded. Thus the cost minimization has yielded a layout where not only have the visual incongruities been avoided, but the majority of the visual interest has been preserved. The resulting layout is the most visually pleasing of the available alternatives.

Figure 5: The result from a different set of candidate layouts

Applying a different set of candidate layouts creates a completely different visual appearance to the final layout. In figure 5 we have applied the constraint that one of the images should be one third the size of the other, and should be positioned completely within the borders of the other image. For this set of candidate layouts, the optimal solution positions the inset image over the area of gravel in the background image, occluding virtually no visual interest.

Figure 6: Application of occlusion costs to single images

Minimization of occlusion cost may be similarly applied to single images. In Figure 6 we see the results from a different set of candidate layouts. In this case the set consists of all possible positions of the image whereby the image is scaled such that it completely fills the window. Since the aspect ratios of the images and the window are different, this will necessarily lead to some parts of the images being cropped. By applying the method described above, the optimal solution is to crop each image according to the green box. This will minimize the occlusion of interest within the image, and in both cases leads to a visually pleasing layout with no visual incongruities.

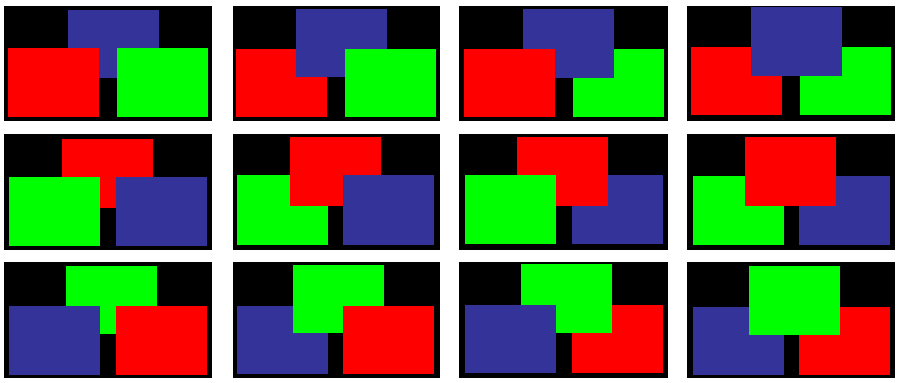

Figure 7: A candidate set of layouts containing three images

Although not described further in this paper, the occlusion cost may be applied to candidate layout sets of three or more images. Figure 7 shows a trivial set of layouts containing three images. As before, each may be positioned in any of the three positions, and can either overlap or be overlapped by the other images.

3. AREA-COMPENSATED OCCLUSION COST

3.1. Scalable images

In the examples detailed above, the set of candidate layouts has applied a fixed size to each of the images in the layout. This significantly reduces the population of the candidate set, but also often results in a set where no candidate layout sufficiently reduces the visual incongruities, and so the final layout will not be acceptable.

Figure 8: Two images, and an optimal, but unsatisfactory image layout

Figure 8 shows a pair of images laid out according to the candidate set in Figure 2. The area being occluded consists mainly of gravel, which has little visual interest. However, the occlusion of the aircraft wing has created a visual incongruity which yields an unsatisfactory layout. While the wing itself is correctly attributed high saliency in the saliency map, it is outweighed by the large area of low saliency immediately surrounding it. Rearranging the layout of the photos to expose this area of the image will necessarily expose the large area of low saliency, and occlude some other area which may be more interesting.

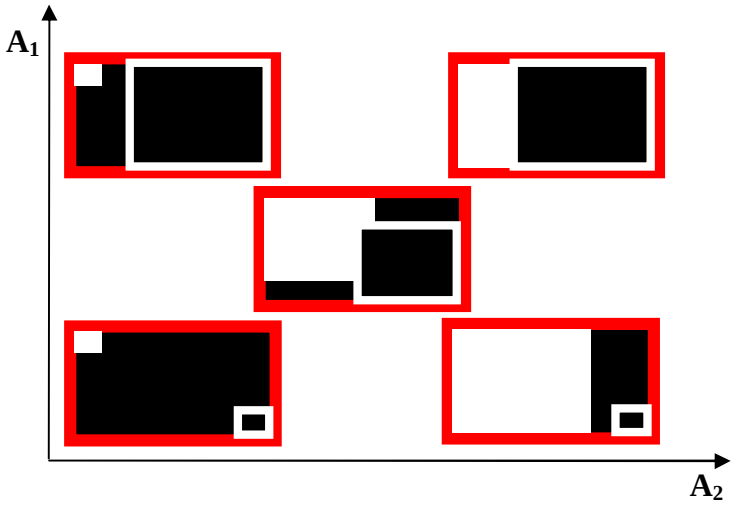

Figure 9: Graphical representation of a candidate image set with scalable images

The solution is to create a new set of candidate layouts in which the images are permitted to scale independently in each of the either configurations, as shown in Figure 9. This figure represents a continuum of candidate layouts where the area of image 1 (A_1) and the area of image 2 (A_2) are permitted to vary independently. Note that the layout in the centre indicates a special case where $A_1 = A_2$, which is the basis for the layout in Figure 8.

3.2. Area compensation

For the set of candidate layouts in figure 9, the basic form of occlusion costing is insufficient. Since the cost is independent of the size of the images, the optimal solution will inevitably have both images scaled down below the point where S_V, the visible saliency, is equal to S_T, the total saliency. In this continuum of solutions, there is no image occlusion, but there is significant wasted space within the frame.

To compensate for the area of the images, we must extend the form of the occlusion cost function. The new form of the occlusion cost for each image, i, is a function of three variables: the fraction of saliency which remains visible (as before), the total saliency density within the frame, and the saliency distribution.

$$F_S(i) = \frac{S_V(i)}{S_T(i)}, \quad \rho_S(i) = \frac{S_V(i)}{\sqrt{A_w}}, \quad D_S(i) = \frac{\sqrt{A_i}}{S_T(i)} \tag{2}$$

The fraction of visible saliency, F_S, is equivalent to the basic form of the occlusion cost, but in this case is calculated for each image individually. As before, this term will be maximized when all of the saliency of all the images is visible, that is when no part of any image is occluded.

The total saliency density, ρ_S, is simply given by the amount of visible saliency, divided by the linear scale of the window, which is denoted by the square root of its area, A_w. This term seeks to maximize the amount of total visible saliency within the boundaries of the window, which directly results in an increase in visual interest of the final layout.

Finally the saliency distribution, D_S, depends on the total saliency of the image and the square root of its area, A_i. This term serves to scale each image with respect to its total saliency. For images with a high saliency, an even distribution will be maintained when that image is scaled up, and others are scaled down to compensate.

Taking the reciprocal of these three terms and multiplying them together yields the component of the layout cost attributable to each image, i:

$$C(i) = \frac{1}{r_S(i) * F_S(i) * D_S(i)} = \left(\frac{\sqrt{A_f}}{S_V(i)} \right) * \left(\frac{S_T(i)}{S_V(i)} \right) * \left(\frac{S_T(i)}{\sqrt{A_i}} \right) = \frac{\sqrt{A_w / A_i}}{F_S(i)^2} \quad (3)$$

This shows how the basic form of the occlusion cost function is modified to include the area-compensation factor. Increasing the fraction of visible saliency will decrease the cost function, but that increase is traded off against the size of the image with respect to the window. Finally, the overall cost for layout L, C(L), is simply the geometric mean of the costs for each image.

$$C(L) = \sqrt[n]{\prod_{i=1}^{n} C(i)} \quad (4)$$

These two equations constitute the final form of the area-compensated occlusion cost function.

3.3. Continuity of solution set

Since we are relaxing the constraints on the relative scales of the images, our solution set is now continuous in two dimensions. While this conceptually leads to a continuum of possible layouts, the set is effectively limited by the quantization of the output display, or of some manually determined quantizing factor. In either case, the set of candidate layouts will be many times larger than the trivial set illustrated in Figure 2, and may approach tens or hundreds of thousands of candidates. However, even this extended set does not depict every possible configuration of two images within the frame.

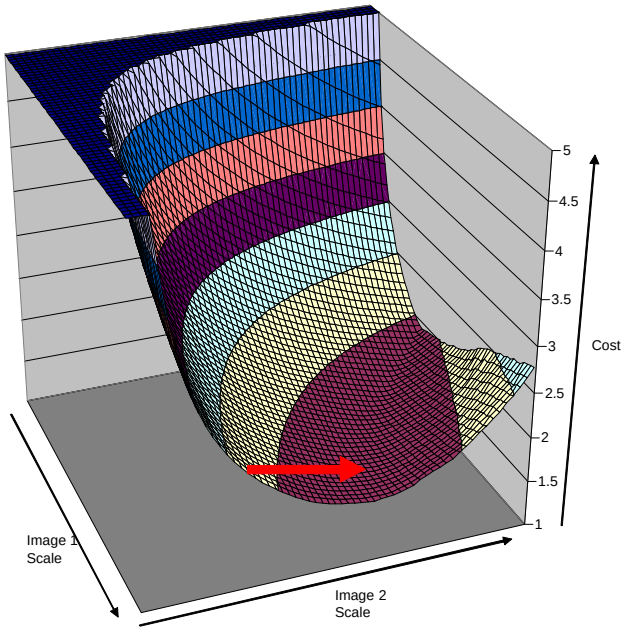

Figure 10: A continuous two-dimensional cost surface

Applying the area-compensated occlusion cost to the images in figure 8, and the candidate layout set in figure 9, yields the two-dimensional cost surface shown in Figure 10. Calculating the cost function at each point on this 2D surface can be very computationally intensive, and so we utilize a method of steepest descent to more quickly determine the minimum cost value.

What cannot be immediately discerned in figure 10 is that the cost function is not a smooth surface. Since it relies on the distribution of saliency in both images, which is not evenly distributed, the cost function has low-amplitude, high-frequency component. Thus a simple steepest-descent will tend to find one of the many local minima, which are not indicative of optimal layouts. To circumvent this problem we apply a recursive steepest descent with a decreasing step size. Starting with a large step size of 100 pixels (the total viewing window in this case is 480 pixels tall), the minimum is found. Using this as a starting point, the step size is reduced to 50 pixels, and the steepest descent is applied again. This dramatically reduces the number of computations required, from about 200,000 when calculating the entire surface, to less than 100 in most cases. Creating a slideshow of 100 images (50 image pairs) can then be performed in less than 7 seconds on a 3GHz PC.

As before, the minimum of the cost function, indicated by the red arrow in figure 10, indicates the scales which should be applied to both images. The optimal layout can then be constructed by applying those image scales and compositing as before.

3.4. Results for the area-compensated occlusion cost

Using the method described above, we can determine the optimal layout for the images shown in figure 8, with the set of candidate layouts show in figure 9.

Figure 11: The optimal layout of the images in figure 7 with the set of candidate layouts in figure 8.

The optimal layout is shown in figure 11. Note that in contrast with the layout in figure 8, the wing of the aircraft is now visible, and only areas with very low visual interest are occluded. Since the visual incongruity has been eliminated, this is a much more attractive layout for the two images.

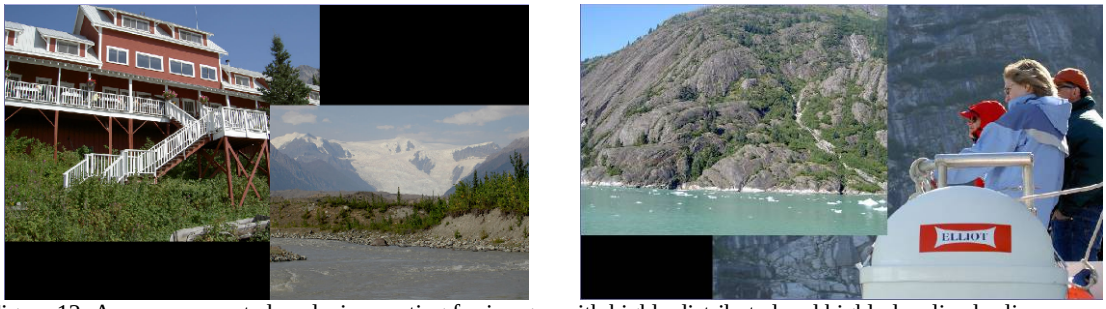

Figure 12: Area-compensated occlusion costing for images with highly distributed and highly localized saliency maps

One key feature of the form of the area-compensated occlusion cost is that it does not preferentially scale one image over the other, but will tend to keep both images approximately the same size. This is illustrated in Figure 12. The layout on the left contains two images with very even distribution of saliency across the frame. An even distribution means that there is a high level of ambiguity as to which areas of the image are of high interest, and which can be safely occluded.

In this situation, the optimal layout minimizes the overlap between the images, in order to minimize the chance that an area of high interest is occluded. However, rather than scaling one image up and shrinking the other dramatically, the output has both images of almost the same size.

The image on the right in figure 12 shows the opposite effect. The image being occluded has very localized saliency, where almost all the interest is concentrated in right hand half of the image. This means that both images can be scaled up to relatively large sizes without occluding any areas of interest. The result is a layout which utilizes a very large fraction of the available window space, which is a very desirable result.

Figure 13: Area-compensated occlusion costing for single image layouts

As with the basic form of occlusion costing, there is no limitation to the number of images in a given set of candidate layouts. Figure 13 shows there results when the set of layouts includes only a single image, which is permitted to vary in both scale and position within the window, here denoted by the green box. Note that the image is allowed to scale both significantly smaller and significantly larger than the area of the window. The minimization of the cost function here depends on the trade-off between increasing the image size, and capturing the largest quantity of saliency within the window. This correlates to a trade-off between the saliency distribution and visible saliency fraction terms in the area-compensated cost function. For the image on the left, the even distribution of saliency across the image window weights the cost function towards a smaller image size, so more saliency can be captured within the window boundary. The image on the right eliminates a large area of low interest towards the bottom and edges of the image, and permits the image to fill the window entirely.

4. EXTENSIONS TO THE OCCLUSION COST MODEL

There are a large number of extensions to the occlusion cost model described in this paper. A select few of those are described below, but the full spectrum of applications is left to the imagination of the reader.

4.1. Image motion

In all the above examples, we have limited discussion to layouts where images are displayed in a static fashion. A sequential presentation of image pairs, for example, succeeds in increasing the visual interest beyond a sequence of static images, but will eventually become repetitive. One way to avoid this is to add movement of the images within each layout. While small movements of each image are unlikely to result in any significant visual incongruities, large movements may apply occlusion costing with a trajectory-based approach. Calculating the cost function at each point on a candidate trajectory and averaging out the results will give a reasonable cost value for that trajectory.

4.2. Clip Art

A number of slideshow packages allow the insertion of clip art, captions, and other graphics to a series of images. Treating these graphical elements as multimedia objects in the same manner as before, they may be automatically superposed above a given image in such a way as to avoid visual incongruity.

4.3. Video clips

In a similar fashion to dynamic image motion, occlusion costing may be applied to video clips within a slideshow. Calculating the average saliency of each pixel over the length of the clip, for example, gives a first approximation to a saliency map for the video. This may be applied in precisely the same way as a static image. When the clip is in motion relative to the frame, and to the other objects within the frame, a trajectory cost may be calculated in a manner similar to that described for moving images above.

4.4. Print-based layout

There is no limitation within this model that the output should be to a screen in the form of a slideshow. Laying out images in a partially overlapping model is equally applicable to photo albums, catalogs, or any other product containing multiple images within a fixed layout area.

5. CONCLUSIONS

Presenting images in a visually interesting manner is an increasingly valuable proposition. Current technologies for slideshows or print-based materials rely on a variety of techniques including image rostrum and automatic cropping techniques to enhance the experience. Occlusion costing, in both basic and area-compensated forms, provides a method to automatically create much more complex and aesthetic arrangements of images, while avoiding the visual incongruities that would inevitably result from the generation of a random layout. It can be applied to a wide variety of image layout applications beyond the slideshows described in this paper.

Occlusion costing has been successfully applied to groups of one, two and three images with a wide variety of candidate layouts. It is highly reliable, and the few cases in which it fails are usually due to inaccurate analysis through either the saliency generation or face detection algorithms.

ACKNOWLEDGEMENTS

I would like to thank Maurizio Pilu, who provided the implementation of the saliency map generation.

REFERENCES

1. X Hua, L Lu, H Zhang, *Automatically Converting Photographic Series Into Video*, Proc. 12th ACM Conference on Multimedia, October 2004
2. S Drucker, C Wong, A Roseway, S Glenner, S De Mar, *MediaBrowser: Reclaiming the Shoebox*, Proc. ACM Working Conference on Advanced Visual Interfaces, May 2004
3. C B Atkins, *Adaptive Photo Collection Page Layout*, Proc. of the IEEE International Conference on Image Processing, Singapore, October 2004
4. L Itti, C Koch, E Niebur, *A Model of Saliency-Based Visual Attention for Rapid Scene Analysis*, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol 20, No 11, November 1998
5. P Viola and M Jones, *Robust Real-time Object Detection*, 2nd Intl. Workshop on Statistical and Computational Theories of Vision – Modeling, Learning, Computing and Sampling, Vancouver Canada, July 2001