텍סיית-מד媳ר המונגש לארישר חככיה עלםדה-דוקטור

תאריך: פברואר 2010

Vladimir (Zeev) Farbman - פרבמנ

Prof. Dani Lischinski - דני לישקין

נושאים העסיקה:

שימור באלגוריתמים שלرأיות מומחשים בגרפיק

High Level Image Understanding For Computer Graphics

אישור המדריך:

הריני מ أبرיא את הנושא ואת החככיה, מסכמים להדיך את המומחה בעובדיה ו.
Research Program

1 Introduction

In recent years we see an explosion of digital media content over wide range of platforms, from personal computers and game consoles to mobile phones and PDAs. Decreasing costs of video cameras and their integration into mobile devices contributed to success of such digital media sharing platforms such as "Youtube" and "Flickr". Despite the increasing popularity of digital media and steady improvements in computer hardware, creating high quality content with a potential artistic or/and commercial value is still a laborious and time consuming process, as indicated by the sky-rocketing increase in the costs of asset creation for feature films and computer games. The high availability of the digital media and the difficulty of creating high quality content contributed to the long standing motivation of the researchers in the field of computer graphics community to create interactive tools which are aimed to both automate a laborious parts of the content creation process and create a fundamentally new ways to interact with digital media.

The goal of my research so far was to find a ways to leverage both the computational power of today's computers and the intelligence and the creativity of their human operators by combining intuitive, easy-to-use user interfaces with sophisticated and powerful algorithms capable of translating the user's intent into the desired outcome.

In this research proposal I will outline both some immediate promising directions which stem from my previous work and more ambitious, stretch goals, partial achievement of those can be considered as a significant contributions to the state-of-the-art in the computer graphics and interactive techniques.

2 Scientific Background

For quite some time now, significant part of the research in computer graphics has focused on providing intuitive user-interfaces, backed up by powerful algorithms, to make common tasks that graphics artists, digital photographers, and also ordinary users are faced with less tedious.

Below I briefly summarize the background and the recent work most related to the goals of my current research.

2.1 Edge Aware Image Editing

Recently, there has been a lot of interest in so called edge aware image editing. In the most general terms, the unifying idea of the related works is to allow the user carry out different adjustments in different
parts of the image in such a manner that the final result will appear visually plausible. Such tasks in the existing commercial software require the user to specify a mask which will indicate the region where the manipulation should be applied. Edge aware image editing aims to minimize the amount of input from the user and either implicitly or explicitly constructs a mask based on the input and the image in hand.

In the pioneering work, Levin et al. [14] introduced an optimization framework for colorizing grayscale images by propagation from a set of sparse user-provided constraints. This approach was generalized to other tonal manipulations [16], as well as to natural image matting [15].

Another tightly related set of methods is based on edge-preserving smoothing, which became valuable tool for a variety of applications in computer graphics and image processing. The common denominator of those methods is the need to decompose the image into fine and coarse variations in such a manner, that both parts can be manipulated separately and then recombined, yielding a visually pleasing result.

Examples of such applications are: HDR tone mapping, flash/no-flash image fusion, transfer of photographic look, image editing, and for other tasks.

All those methods assume the notion of edges - discontinuities in the image which indicate a transition form one region to another. Those transitions are measured as simple difference in luminance or color between adjacent pixels. First research directions described in this proposal aims to define a new distance metric between the pixels which captures better our intuitive notion of edges.

2.2 Seamless Compositing

Some creative manipulations require to copy parts of one image into another image. In many cases we will be interested in "concealing" the fact that the result is a collage of different sources. One example of such manipulation is stitching images together in order to create a compelling panoramas with very wide field of view. Another example is copying an object of interest from one image and pasting it into a different image.

Blending and compositing techniques were used for decades to carry out such tasks and recently there was a lot of interest in gradient domain techniques, which operate directly on the gradient field of an image [21, 14, 2, 17].

Psychologists have long discovered that the human visual system is much more sensitive to local contrasts than to absolute luminances or to slow changes in the luminance [13, 20]. In particular, slow luminance changes, which are suppressed by the human visual system as part of lightness constancy, may be often superimposed over an image without a noticeable effect.

Gradient domain methods take advantage of the above properties, and modify images by manipulating their gradient field to perform a variety of tasks, ranging from shadow removal [25, 8], to tone mapping [7], seamless stitching [14, 2], image cloning [21, 10, 12], seamless video editing [24], and, recently,
gradient domain painting [17].

Reconstructing a new image from the modified gradient field typically requires solving the Poisson equation, which yields the image whose gradient field is closest (in the $L^2$-norm sense) to the modified one, subject to some boundary conditions. For example, in Poisson cloning [21], the gradient field (sometimes referred to as the guidance field) inside the cloned region is taken from the source image, while the values of the target image along the boundary of the cloned region are used to define the Dirichlet boundary conditions for the equation.

Solving the Poisson equation for large images is a computational and memory intensive task. Agarwala [1] observed that in the case of gradient domain stitching, one essentially solves for an offset function that is smooth away from the seams. This makes it possible to obtain an accurate solution by constructing a reduced linear system using an adaptive quadtree subdivision of the domain. This method has been shown to be significantly faster and more scalable than general Poisson solvers for stitching large images.

McCann and Pollard [17] describe a fast GPU implementation of a multi-grid Poisson solver, with which they achieve real-time interactive performance for gradient domain image editing operations, including seamless cloning. While their system outperforms previous methods, it does involve a substantial memory footprint, and the authors report that performance drops down once this footprint exceeds the available video memory.

Solving the Poisson equation with Dirichlet boundary conditions, as is the case in the context of seamless cloning, in effect smoothly interpolates the discrepancies between the boundary of the source patch and the target across the entire cloned area. In Farbman et al. [5] we introduced an alternative, coordinate-based approach, where rather than solving a large linear system to perform the aforementioned interpolation, the value of the interpolant at each interior pixel is given by a weighted combination of values along the boundary. More specifically, our approach is based on Mean-Value Coordinates (MVC) [9, 11]. The use of coordinates is advantageous in terms of speed, ease of implementation, small memory footprint, and parallelizability, enabling real-time cloning of large regions, and interactive cloning of video streams. One of research directions described in this proposal is concerned with various extensions of this approach.

3 Research objectives & expected significance

As mentioned earlier, I see as the longer term objectives of my research are to develop new tools for performing graphical operations on still images and video. More specifically, and in the shorter term, I plan to focus our research activity on two research areas described below. The unifying chord of both those direction is to try and integrate the state-of-the-art computer vision algorithms in computer graphics tools.
One area is concerned with the development of new metric for measuring distances between pixels, which better differentiates between edges and texture. Another area is concerned with the development of novel tools for seamless compositing of still images, as well as video. In all of these projects the emphasis is put on interaction, as I believe that the most effective tools are obtained by leveraging the strengths of both the artist and the computer: the artist should be free to focus on her creative goals, while the computer takes care of as many tedious aspects of the process as possible. These proposed research directions are described in more detail in the next section.

The significance of the proposed research stems from the increasing importance that computers and software tools play in the creative workflow of visual artists and animators. In particular, with the increasing abundance of digital imagery, due to the proliferation of digital still and video cameras, I believe that the need for intuitive and effective image and video manipulation tools will only continue to grow. With my research I hope to take part in pushing the envelope of what creative artists, as well as “ordinary people” are able to achieve using a computer.

4 Detailed description of the proposed research

4.1 Pixel Affinities for Edge Aware Image Editing

As we already hinted in the background section, edge aware image editing, which has recently emerged as useful tool for a variety of image editing and manipulation tasks, requires a similarity measure (sometimes referred to as affinity) between pairs of pixels. These pairwise affinities dictate how pixels are averaged together or how values are propagated from one pixel to another. It is assumed that a high difference in the pixel feature vector values (typically rgb or other color space values) correspond to a significant changes in the image - edges. Smaller values correspond to similar values which should be “treated/clustered together” in the smoothing/interpolation process. The affinities used in practice are typically a simple monotonic function of the difference between some low-level feature vectors at each pixel, which can be as simple as the pixel’s luminance or its coordinates in some color space.

Such definition of affinities can fail in two significant cases. First is the presence of texture, ie a region where per-pixel intensity difference may be quite high, but still representing a region in the image which we would like to be treated uniformly. Second problem is that a low contrast edges between have a small rgb difference.

A promising direction of my ongoing research is to define the distances between the pixels by looking into the actual distribution of the pixels feature vectors. Based on the observations of the researchers in the computer vision community, we claim that in natural images, feature vectors corresponding to the same surfaces tend to span low-dimensional manifolds (in potentially very high dimensional feature vector space), unlike feature vectors corresponding to different surfaces. For example, it is well known
that colors of the same surface tend to create prolongated, line-like clusters in tree-dimensional color space. For more discussion see[19].

One idea that showed a lot of promise so far is to use so called diffusion distance, instead of regular euclidean distance. Diffusion distances account for length and density of paths in the data, but unlike current methods used in path-based clustering, lend themselves to faster computation. In addition, diffusion distances have a built-in notion of time, which makes it possible to control the granularity (scale) of the clusters in an efficient and intuitive manner.

Our experiments show that one can use diffusion based distances instead of euclidean color distances, while setting up the edge-aware operators and this way to get more control over the result of the manipulation.

I want to continue and explore both advantages and limitations of diffusion maps in this context. One potentially interesting direction is to see if something can be gained to switching to time non-homogenous Markov chains (since it was shown to be beneficial in some related problems).

While using diffusion distances is very useful in overcoming textured regions, it doesn’t solve a problem of low contrast edges. Let’s see an example, demonstrating why this is a problem for edge aware image editing. Say, the user have an image of dark-colored person on some rather dark background. He wants to make the person to be more discernible from the background. In order to do that, he wants to increase an exposure in the region of the image occupied by the person. One way to convey such manipulation with the current edge-aware methods is to put a rough scribble in the region where the user is interested in applying the change. In our scenario this wouldn’t produce a desired result, since the propagation of the edit (which increases the exposure) won’t stop on the weak edges on the boundary between the man and the background. The result will be that either the final image will have the background exposure as well (over-propagation) or the person will remain under-exposed (under-propagation).

In order to overcome such problems, the algorithm has to have much higher-level notion of an object. One direction to try is to integrate the diffusion maps into higher level algorithms. For example, I want to try and integrate diffusion maps into active contour models. Those models try to find a continuous boundary on an object and thus are can potentially find very low contrast edge or even fill the gaps in a broken edge.

4.2 Seamless Cloning of Images and Video

The problem of interactive seamless cloning was presented in Section 2.2. Our recent work on coordinate-based MVC cloning [5] has made it possible to perform this operation at fully interactive rates on high resolution still images, as well as on video sequences. However, the quality of the seamless cloning results produced with this approach is essentially the same as that of Poisson image editing, which in
turn suffers from a number of limitations. In my research I plan to extend our approach so as to address some of the more restrictive limitations.

One commonly encountered limitation stems from the fact that Poisson and MVC cloning smoothly interpolate the differences between the source and the target images from the boundary of the cloned region across the entire region. Thus, the abrupt discrepancy between the two images is converted to a slowly varying change across the entire region, something that the human visual system is less sensitive to. While this is exactly the right thing to do in some circumstances (such as panoramic stitching of images), in other circumstances this strategy may result in undesirable shifts in overall brightness or unpleasant color casts.

This problem is rooted in the fact that these cloning methods do not take into account the content inside the cloned patch. For example, human visual system is very sensitive to appearance of human faces. Color shift in the region of face may severely impair our perception of the image as being aesthetically pleasing or even realistic.

A stretch goal in this context will be an integration of high level object recognition algorithms, which upon realization that there is a distinct object of interest in the selected region, will suggest the user different cloning options accordingly.

One more immediate goal is to use the edges in the source patch in order to shape the cloning interpolant. We are interested to have a constant interpolant in the areas that are separated from the boundary by strong edges, since we assume that those edges represent an object of interest and we are tolerant to abrupt changes due to edges. One strategy to accomplish this could be the use of geodesic distances, when computing the interpolant. Another possibility is to formulate the problem on kernel-based representation suggest by [6].

It is well known that Poisson image cloning works best in the presence of a smooth boundary. When exists a discrepancy (in scale, orientation etc...) between the texture in the source and target images, the mere fact that there is no seam on the boundary, doesn’t stop the human observer from recognizing the cloned patch as being a part of another image and mentally classifying the result as unsatisfactory or unrealistic. Trying to blend high-frequencies doesn’t solve the problem, since the blended results typically appears blurry and unrealistic. It is interesting to try and think how one can evolve one texture into another, so the in-between results don’t exhibit any unwanted artifacts.

Yet another interesting direction is to explore the possibilities for a new video cloning interfaces. In [5] we showed a rudimentary video cloning tool. One of the problems that we discovered that the difference in a frequency of temporal changes between the source video and target video can create implausible results. Integrating a degree of control over temporal frequency in the interface can potentially overcome those problems.
4.3 Image Understanding in Computer Graphics

Computers became indispensable tools in creating and manipulating visual content. From medical imaging and convensional photography to industrial design and film industry, professionals in all those fields depend on tools developed in computer graphics community.

Yet many tasks are still extremely laborious, which I partially attribute to the fact that higher level computer vision algorithms aren't used to their full extent, while creating computer graphics tools.

Let's take a task of coloring black and white movies, which drew a lot of attention recently. State-of-the-art methods still require a great amount of user input - basically, constraining a color in each smooth distinct region. It still impossible to reliable color video sequences, even if the constraints are set in some key frames.

Much more user-friendly and less laborious interface would be if an algorithm, given a black and white sequence would analyze the scene content and suggest a user possible coloring schemes for a different objects in the scene.

I believe that the integration of higher level algorithms developed in computer vision community with lower level image manipulation algorithms originating in computer graphics and image processing community can make such tools a reality in years to come.

4.4 Materials and Methods

Typically the prototyping and development of the new algorithms and tools is done with state-of-the-art personal computers. Higher level computer languages, such as Matlab and Python are used in an initial stages of the project. In order to have an efficient demonstration, we typically aim to have a C/C++ implementation with GPU support (where applicable). The required mathematical background can vary from project to project, but my current experience indicates that in order to push forward in the aforementioned directions, one need a strong background in applied mathematics, especially in linear algebra and it's connection to signal processing.
Bibliography


