GROUND-TRUTHED VIDEO GENERATION FROM SYMBOLIC INFORMATION

Andrew Lookingbill[1], Emilio R. Antunez[1], Berna Erol[2], Jonathan J. Hull[2], Qifa Ke[2] and Jorge Moraleda[2]

[1]Stanford University
Stanford, CA
{apml, eantunez}@stanford.edu

[2]Ricoh California Research Center
Menlo Park, CA
{berna_erol, hull, moraleda, qifa}@rii.ricoh.com

ABSTRACT

An algorithm is presented that automatically generates ground-truthed video from a symbolic description for an object and a specification for the movement of a handheld video camera around that object. This provides a method to generate large amounts of training and test data for the development of computer vision algorithms. We describe an implementation of this technique for an imaging application in which a cell phone video camera is moved over a paper document. Experimental results demonstrate the similarity of images captured by the real camera to images generated by the proposed technique.

1. INTRODUCTION

Video recognition algorithms, such as face recognition or scene analysis techniques, require large numbers of ground-truthed video clips for their development and validation. Ground truth information can be very detailed and can include the identity and location of relevant regions in every video frame. For applications such as video surveillance of crowds of people, the ground-truthing of a video clip can require the segmentation of each frame into regions associated with every person such as their body parts, their clothing and the objects they carry [9]. If the camera can move, a large number of video clips may be needed to represent the diversity of views that might be encountered.

The typical solution to the production of ground-truthed video requires the manual annotation of every frame – obviously a tedious and expensive process. While there do exist tools that make this easier (e.g. Viper [8]), the cost of producing ground-truthed video is a significant impediment to the development of recognition algorithms. As a result, small sets of ground truth data might circulate among the academic research community (e.g., the PETS dataset for crowd surveillance [9]). Large collections created by commercial organizations are often considered a secret competitive advantage.

This paper describes a novel solution to data acquisition for video recognition development in which we generate video clips from symbolic information so that each frame is *automatically* associated with ground truth information. Furthermore, we incorporate a model for the movement of a camera that allows us to generate video clips for arbitrary paths around an object. This enables us to provide a virtually unlimited number of video clips at almost no cost, thereby allowing any researcher to develop recognition algorithms with an accuracy unhindered by the amount of training data.

2. ALGORITHM

The algorithm for video generation from symbolic information is shown in Fig. 1. Given a path in space, as specified by a series of control points, and a model for the movement of a video camera between those points, a path generator provides a sequence of coordinates for the position of the camera with respect to a given object. One set of coordinates is provided for each time t when the camera would capture a frame. The image generation model uses characteristics of the camera and a ray-tracing algorithm to produce a sequence of frames, one for each set of coordinates. Each frame is distorted to account for a given warp and shadow on the original object. The collection of frames is concatenated to produce a video clip.

In our path generation algorithm, the position and orientation of the camera is represented by three position parameters and three Euler angles. Each of these is updated using a discrete-time linear dynamical system. For example, the position of the camera, with respect to the origin of the source document (in inches) is given by X. The state of the camera's X location at time n is given by:

$$X_n = \ [P[n]\ V[n]\ A[n]\ J[n]]'$$

where P[n] is the position of the camera in the X direction, V[n] is the velocity, A[n] is the acceleration, and J[n] is the jerk. The state of the camera's location at time n+1 is given by the following relation: $X_{n+1} = A^*X_n + B^*u(t)$,

where u(t) is known as the driving force and

$$A = \begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad \text{and} \quad B = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}$$

The study of linear dynamical systems tells us that if a state is reachable in n steps, then the controllability matrix is:

$$C_n = [\ B\ \ AB\ .\ .\ .\ \ A^{n-1}B]$$

If there is one way to reach X_{des} in n steps, there may be many ways to do so. We used the minimum energy solution for getting from a starting point to a desired state:

$$u(t) = C_n^T(C_n\ C_n^T)^{-1}X_{des}$$

A small amount of zero-mean Gaussian noise is added at each step to achieve a realistic trajectory. The smooth acceleration and deceleration as well as the random but continuous movement that are characteristic of a human operator are readily apparent in the generated trajectories. At each control point, we simulate a "hovering" of the camera by providing no explicit driving force. Zero-mean Gaussian noise is added in place of u(t).

1-4244-1017-7/07/$25.00 ©2007 IEEE

3. IMAGE GENERATION

The algorithm that generates individual frames receives extrinsic parameters that specify the position of the camera with respect to the object, a symbolic description for the object that enables the generation of photorealistic images, ground truth that identifies and locates items of interest on the object, parameters that can be applied to distort the generated image (e.g., a specification for the shadow cast by an external object), and intrinsic characteristics of the camera's optics.

An example of a symbolic object description and ground truth is a postscript file that can be processed by a rendering program to generate a raster image that's equivalent to a printed document. The rendering program (e.g., ghostscript) can be modified so that it outputs ground truth information that includes the position and identity of every character in the image.

For our purposes, the intrinsic parameters of the camera are the two focal lengths, f_x and f_y (in pixels), principal point coordinates, cc_x and cc_y (in pixels), the skew coefficient, and five coefficients describing radial and tangential distortions. This is the model used by Bouget in his camera calibration toolbox [6]. This information is used during ray tracing to determine where rays cast through each pixel of the sensor from the camera origin will intersect with the source document.

3.1 Camera-Related Effects

The image generation algorithm models several of the characteristics that have the most significant effect on the images produced by a handheld video camera.

Our model incorporates *sensor noise* as the pixel gain non-uniformity described in [7]. We use a uniform variable, with mean 1, and a range that can be adjusted to achieve the desired level of noise. This gain non-uniformity is multiplied on a per-pixel basis by the result of the ray-tracing algorithm.

In practice, the *range of the histograms* of images captured using a real sensor is smaller than the range of intensity values present in the scene. This effect can be modeled either by mapping the histogram values of the virtual image to fall within the range of values appearing in some sample image taken with a real camera, or by a more complicated histogram matching which attempts to transform the virtual image's histogram in such a way that its cumulative distribution of pixel values matches that of the example image by way of a look-up table transformation.

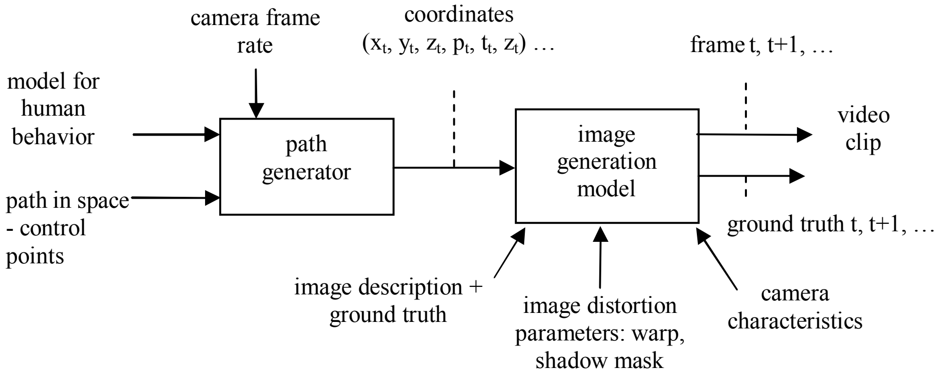


Fig. 1. Algorithm for video generation from symbolic ground truth.

The model implements a *vignetting*-like effect to capture the so-called "cosine-fourth" falloff in brightness as the angle between the ray corresponding to an image pixel and the optical axis of the camera increases by multiplying a pixel's value by the cosine of that angle raised to the fourth power. Vignetting in real images is the result of distant off-axis light rays not reaching the physical aperture due to obstruction by lens elements.

Focus blur was implemented as a single Gaussian point spread function whose sigma is the absolute value of the difference between the distance to the page along the optical axis of the camera and the distance at which the camera was empirically determined to be "in focus" (both in inches). This sigma is then scaled by a user-specified value to control the magnitude of the focus blur. The blur therefore increases and decreases linearly as the camera is moved.

3.2 Environmental Effects

The image generation algorithm also models several environmental characteristics that can significantly alter the appearance of the images a handheld camera produces.

In order to model the large global *shadows* often cast over an object by a user's arm or the camera, a *shadow mask* is used. The mask is multiplied element-wise with the image calculated by the ray tracing algorithm.

An artificial but perceptually believable *motion blur* is achieved by calculating the virtual image at a set number, k, of equally-spaced, intermediate positions that the camera would occupy if it were moving with a certain velocity and had a given exposure time. The final image is the average of these k intermediate images.

To model the effects of material and illuminant properties, a *general lighting model* was implemented that includes (grayscale) ambient, diffuse, and specular components as described in [1]. When the ambient, diffuse, and specular components are added, the total intensity value is used to modulate the amplitude of the corresponding pixel in the virtual image.

Extra information in the form of *background* clutter in a test or training image can cause problems for any object recognition application. In order to mimic this effect, support was added for large, high-resolution images of desktop scenes to be used as backgrounds for the raytracing routine. Ideal images for this purpose have a small depth of field, with the scene plane parallel to the image plane of the camera.

4. IMPLEMENTATION

We implemented a video generation system that simulates the movement of a Treo 700w cell phone video camera between points above the surface of document images. We capture the symbolic description for the document images in the Windows XP print driver pipeline while Microsoft Word is printing. We render a separate bitmap that is guaranteed to be equivalent pixel-for-pixel to the physical paper document. The rendering process also creates an xml file that contains the bounding box and identity for every character glyph in the bitmap.

Fig. 2 shows an example of the implementation. Fig 2 (a) shows a thumbnail for a document with an example path composed of six control points. The first is 14 inches above the document, the second 9 inches above and the third 4 inches above. Several individual video frames are shown in Fig. 2 (b). An example of the ground truth provided for each frame is shown in Fig. 2 (c).

The path generation algorithm receives control points, elapsed times between points, and the frame capture rate of the Treo as input. A Matlab implementation of minimum energy solution of the linear dynamical system described earlier generates the position of the camera as 6-tuples (x, y, z, θ_x, θ_y, θ_z) for each instant in time when the camera would capture an image as it is moved between control points.

5. EXPERIMENTAL RESULTS

The success of our methodology is determined by the similarity of the *virtual images* we generate to the *real images* captured on a Treo. Ideally, the images could be the same pixel-for-pixel. But we recognize this could be an unrealistic goal and instead strive for similarity in a relevant feature space. Since effects such as blur, noise, and lighting result in significant differences between the frequency signatures of the real and virtual images, we developed an image distance measure that uses a frequency domain-based feature. Fig. 3 shows an example of a real image from a Treo and the corresponding virtual image of the same document as generated by our system. Fig. 4 shows two examples of virtual images from close-up views of the document.

Our *Neighborhood Frequency Distribution (NFD)* feature works as follows. An image is divided into 8x8 blocks. The 2D FFT of these blocks is computed. Based on the DC coefficient, the block is discarded if it contains either no text, or is all black (corresponds to being inside the boundaries of a printed character). In this way, only blocks on the edges of text characters are used, where the frequency information is the most interesting. For each of the remaining 15 values in the upper left corner of the 2D FFT data, a bit in a 15-bit feature vector is set to one if there is significant frequency content at that location. Otherwise the bit is set to zero. Significant frequency content is defined as being larger than the average value of that component of the 2D FFT taken over a set of representative images.

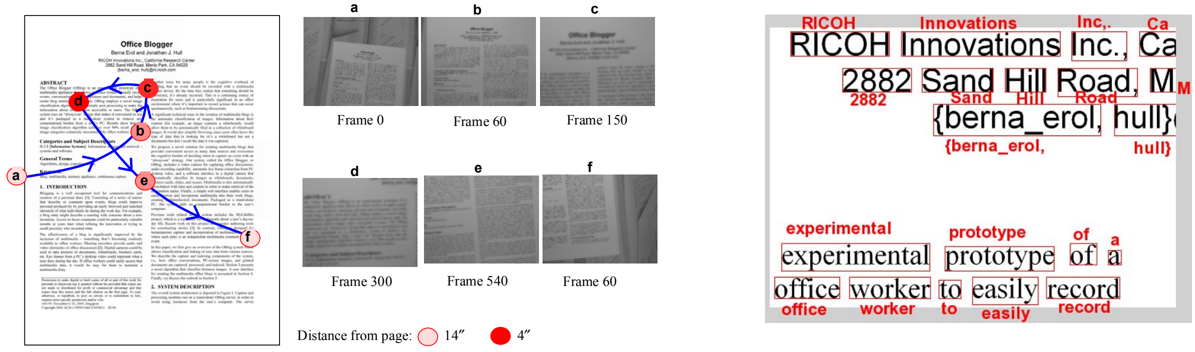
The *distance* between two images is defined as follows. The NFD feature vectors for an image are treated as a population of numbers in the range [0 – 2^15]. The image distance is the ks-statistic from the Kolmogorov-Smirnov test. As the two sets of NFD values look more and more like they were drawn from the same parent distribution this value will tend towards zero.

The experimental evaluation used three sets of images. R1 contained 100 real images of a single document page, each 176x144 pixels in size, captured with a Treo 700w. This set of images has a large variation in camera position and viewpoint.

V1 contained 100 virtual images that were created to simulate a variety of views that the Treo would capture of the same page. These images were created using ray-tracing and a high-quality ground-truthed representation of the source document. These images included the simulated effects of shadow mask, background clutter, motion blur, focus blur, sensor noise, and vignetting. Each of these images was 176x144 pixels in size.

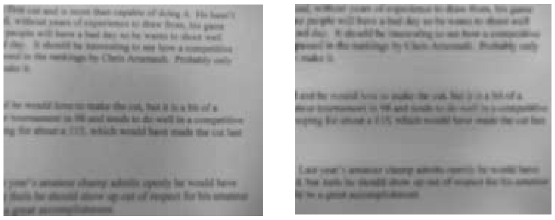
V2 contained 100 virtual images that were created using only the ray-tracing procedure described above, without the addition of the simulated effects. That is, V2 contained high quality non-degraded images. Fig. 3 shows an example from each image set.

The image distances between all the members of R1 and all the members of V1 were calculated (10,000 combinations). The mean of these distances was 0.5564, and the standard deviation was 0.2574. A graphic representation of all the distances (with black


Office Blogger
a
b
c
d
e
f
a
b
c
Frame 0
Frame 60
Frame 150
d
e
f
Frame 300
Frame 540
Frame 60
Distance from page: 14″ 4″
RICOH Innovations Inc,. Ca
RICOH Innovations Inc., Ca
2882 Sand Hill Road, M
2882 Sand Hill Road M
{berna_erol, hull}
{berna_erol, hull}
experimental prototype of a
experimental prototype of a
office worker to easily record
office worker to easily record

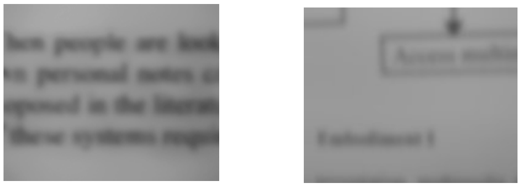
(a) (b) (c)

Fig. 2. Example of video generation. The the original document as captured by the printer driver and showing the control points and path over the document (a), selected frames at the indicated distances from the document (b), and the ground truth provided for each frame (c).



(a) (b)

Fig. 3. A real image a document (a) as captured by a Treo video sensor and (b) a virtual image of the same document.

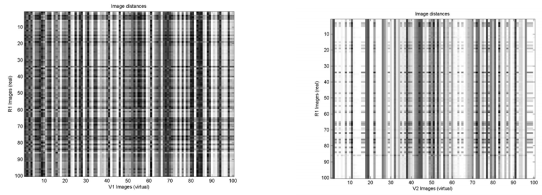


(a) (b)

Fig. 4. Virtual images at 176x144 showing close-up views.

corresponding to a distance of 0, and white corresponding to a distance of 1.0) is shown in Fig. 5. For comparison, when the members of R1 were compared to each other, the mean distance was 0.2601 with a standard deviation of 0.1272.

The image distances between all the members of R1 and all the members of V2 were also calculated. The mean of these distances was 0.8307, and the standard deviation was 0.1696. A graphic representation of all the distances (with black corresponding to a distance of 0, and white corresponding to a distance of 1.0) is shown in Fig. 5 (b). By inspection of the gray scale distribution of the distance plots, we see that the images captured on the Treo are much more similar to virtual images that were generated to simulate the Treo (Fig. 5 (a)) than they are to virtual images that contained no camera effects (Fig. 5 (b)). This is reflected in the significant differences in the mean of the two distributions (0.5564 vs.0.8307).



(a) (b)

Fig. 5. (a) Image distances between real images R1 and virtual images V1 that simulate a Treo. Black indicates distance 0 and white is distance 1. (b) Image distances between V1 and clean virtual images V2. The horizontal and vertical axes identify individual images.

6. RELATED WORK

The automatic generation of ground-truthed data for video recognition algorithm development can provide fertile ground for exploring new methodologies. As with many developments in this field, the work reported here is similar to several previously described ideas and yet appears to be different from any of them. For example, one notable project generated surveillance videos of arbitrary complexity from a collection of manually ground-truthed clips captured with a single camera in a fixed position [4]. This contrasts sharply with our emphasis on fully automatic ground truth generation and the use of a portable video imaging device.

The distortions introduced by an image sensor were modeled in work on document image degradation that simulated a relatively high-resolution flat bed image scanner and advocated the production of large collections of synthetically generated images [2]. Recently, this methodology was extended to incorporate ground truth generation [10]. However, this line of work did not consider a portable video capture device.

7. DISCUSSION AND CONCLUSIONS

We described a novel technique for generating ground-truthed video from a portable sensor that unifies concepts from graphics and computer vision to solve a significant problem in multimedia recognition. Our first implementation addresses a domain (document imaging) in which the undistorted input data is guaranteed to be equivalent bit-for-bit with the corresponding real-world paper document because we capture the data for rendering the document in the print driver. We degrade those images to simulate the output of a cell phone video camera. Experimental results showed that the model produced images that are very similar to those captured by a real cell phone.

Future work should consider applications to multimedia recognition problems such as face recognition in which researchers are striving to produce photorealistic images from three-dimensional models and where there is an almost unquenchable thirst for unlimited amounts of arbitrarily detailed training data. Our technique has the potential to satisfy this need and enable substantial improvements in performance.

REFERENCES

[1] T. Akenine-Moller and E. Haines, "Real-time rendering," A. K. Peters, Natick, MA, 2nd edition, 2002, pp. 70-84.

[2] H. Baird, "The state of the art of document image degradation modeling," Proc. 4th IAPR Workshop on Document Analysis Systems, Rio de Janeiro, Brazil, pp. 1-16, 2000.

[3]. E.H. Barney Smith and T. Andersen, "Text Degradations and OCR Training," Int. Conf. on Document Analysis and Recognition, Seoul, Korea, August 2005.

[4]. J. Black, T. Ellis, and P. Rosin, "A Novel Method for Video Tracking Performance Evaluation," IEEE Workshop on Visual Surveillance and Performance Evaluation of Tracking," 125-132, Nice, France, Oct. 2003.

[5] J. Black, T. Ellis and D. Makris, "A hierarchical database for visual surveillance applications," IEEE Int. Conf. on Multimedia and Expo (ICME), 1571-1574, Taipei, Taiwan, June 27-30, 2004.

[6] J. Bouget, "Camera Calibration Toolbox for Matlab", http://www.vision.caltech.edu/bouguetj/calib_doc

[7] R. Costantini and S. Susstrunk, "Virtual Sensor Design," Proceedings of the SPIE, v. 5301, pp. 408-419, 2004.

[8]. D. Doermann and D. Mihalcik, "Tools and Techniques for Video Performance Evaluation," Proc. Int. Conf. on Pattern Recognition (ICPR), Barcelona, Spain, 4167-4170, Sept. 2000.

[9] V. Manohar, et al., "PETS vs. VACE Evaluation: A Comparative Study," 9th IEEE Workshop on Perf. Evaluation of Tracking and Surveillance (PETS), New York, 1-6, June 18, 2006.

[10] G. Zi and D. Doermann, "Document image ground truth generation from electronic text," IEEE Int. Conf. on Pattern Recognition, vol. 2, 663-666, Cambridge, U.K., Aug. 23-26, 2004.