Animating Cartoons: Exploring the 3rd Dimension

Thesis Proposal

by

Saurabh Garg
(HT026425W)

under the guidance of

Dr. Ng Teck Khim

School of Computing
National University of Singapore
July 2004
## Contents

1 Introduction .......................................................... 1
   1.1 Objective of Research: Adding the 3rd Dimension .............. 1
   1.2 Overview of the Proposed System ................................ 2
   1.3 Organization of the Report ...................................... 3

2 Literature Survey .................................................... 4
   2.1 3D Modeling Techniques ........................................ 5
       2.1.1 Challenges ............................................. 6
       2.1.2 Teddy ............................................... 7
   2.2 3D Motion Capture Techniques ................................ 7
       2.2.1 Challenges ............................................. 8
       2.2.2 2D Tracking ........................................... 10
       2.2.3 Pose Estimation ......................................... 11
       2.2.4 3D Tracking ........................................... 12

3 The Proposed System ................................................ 14
   3.1 3D Object Modeling Tool ..................................... 14
   3.2 3D Motion Capture Tool ....................................... 15

4 Results ........................................................................ 16
   4.1 Modeling .......................................................... 16
   4.2 Pose Estimation ................................................ 16
   4.3 Tracking .......................................................... 19

5 Conclusions .................................................................. 21
Abstract

Development in computer graphics especially photorealistic rendering and availability of faster hardware have made computer generated 3D animations like Finding Nemo, The Bugs Life, Toy Story, Shrek, and Shrek 2 a practical reality. These 3D animated movies look realistic and appealing since computer can create physically correct modeling, lighting and shading effects. 3D animated movies, however, are only a recent phenomenon. Over past 80 years or so many wonderful work in cartoons, such as Popeye, Tom and Jerry have been created. These movies have been entertaining people of all ages from all parts of the world. Success of 3D computer animations and popularity of traditional animations is the main motivation behind this work. By ‘modernizing’ traditional animation with new looks and feel while preserving original story and actions we can enjoy these movies once again and simultaneously preserve original art.

The aim of this work is to research on a set of tools that can be used for converting an existing 2D cartoon video into a 3D environment, where the viewer has the choice of changing the viewpoint. Converting the entire video with all the fine details will be a computationally and economically infeasible task. The idea is to use the original story and sound tracks, model the main characters while replacing the irrelevant details with enhanced visual appearance and new background without affecting the original storyline.
There is no particular mystery in animation...it's really very simple, and like anything that is simple, it is about the hardest thing in the world to do. Bill Tytla at the Walt Disney Studio, June 28, 1937. [11]

1 Introduction

Cartooning is a unique expressive art, which have found its place in our daily life appearing in a lot of places like newspaper columns, television advertisements, movies, and cartoon films. The first animation produced was “Humorous Phases of Funny Faces” by J. Stuart Blackton 1906, where he drew comical faces on a blackboard, photographed them, and then erased it to draw another stage of the facial expression. In the early days animations were created entirely by hand. Each frame was drawn individually (which differs from the previous frame only slightly), painted with colors and converted into film with the help of a camera. Some notable examples are Tom and Jerry, Bugs Bunny, and Mickey Mouse. In this report such animations will be called Traditional Animation.

After the advent of computers people started to automate the laborious and time consuming process of creating animation by hand. Initially computers were used to fill the colors and automatically generate in-betweens. At present all of the tasks can be done by computer except sound recording and drawing of few keyframes. In addition we can add special effects like fire, water etc. Some notable examples are Beauty and the Beast, Lion King, and Prince of Egypt. In this report we will call such animations as 2D Computer Animation.

Development in computer graphics especially photorealistic rendering and availability of faster hardware have made computer generated 3D animations like Finding Nemo, The Bugs Life, Toy Story, Shrek, and Shrek 2 a practical reality. These 3D animated movies looks realistic and appealing since computer can create physically correct modeling, lighting and shading effects. In addition by using computer we can create advanced special effects like fur, cloth, water, crowd and extras or background character simulation which enhance appeal of these movies. In this report we will call such animations as 3D Computer Animation.

Success of 3D computer animations and popularity of traditional animations is the main motivation behind this work. By ‘modernizing’ traditional animation with new looks and feel while preserving original story and actions we can enjoy these movies once again and simultaneously preserve original art.

1.1 Objective of Research: Adding the 3\textsuperscript{rd} Dimension

The aim of this work is to research on the development of a set of tools that can be used for converting an existing video footage of a 2D cartoon film into a 3D computer animation,
where the viewer has the choice of changing the viewpoint. Converting the entire video with all the fine details kept will be a computationally and economically infeasible task. The idea is to keep the original story and the sound tracks, model the main characters while replacing the irrelevant details with enhanced visual appearance and new background without affecting the original storyline. For example if in the original movie Popeye is fighting with Bluto in a courtyard with house in the background, it will make no difference in story if in target movie he fights with Bluto in the field with trees in the background.

Creating an animation either by hand or using computer involves mainly two steps - modeling an object (defining how objects looks like - shape, size, weight and properties like elasticity, rigidity etc) and motion creation (defining how the object will move in space). Based on these two steps we have identified four tools, necessary for converting a video footage of 2D cartoon film into a 3D computer animation.

- 3D Object Modeling Tool
- 3D Motion Capture Tool
- Facial Expression Animation Tool
- User Interface

In this research we will be working on 3D object modeling and motion capture tool, along with their user interfaces. Facial animation is essential for a full animation system, but including it will make scope of this work too extensive. Following section gives the architecture of the system and a brief overview of each tool.

1.2 Overview of the Proposed System

Figure 1 shows the architecture of our system. The input to the system is the digitized video of a traditional cartoon film. The goal of the 3D object modeling tool is to extract the 3D model of the object from a single camera video, some minimal user input will be required. The process of creating a model from the input video is done offline once for each character. The extracted model is then used to track object in the 3D motion capture tool, this is an online process meaning some user interaction is required for all input videos. The 3D pose of the model will also be computed at each frame of the sequence.

Additionally for cartoons, in order to maintain the illusion of being real, it is important that idea being expressed is reflected by both body language (i.e. gesture) and facial expressions. If the body movements of character exhibit anger, a smile on its face will look too awkward and break the illusion. There are two ways of modeling facial expressions: recognize the gesture of the body and use corresponding facial expression from a standard
library automatically; or let the animator decide what type of expression he wants and model the expression from a standard library. This area of 3D animation will not be explored in this thesis work.

1.3 Organization of the Report

The report is organized as follows: the literature survey is presented in section 2. Section 3 describes the approach we will be following. In section 4 some preliminary work is discussed, conclusions are given in section 5.
2 Literature Survey

Lot of work has been done on development of the techniques useful for creation of lively and convincing tradition animation as well as 2D and 3D computer animation. Techniques in 2D computer animation involves application of image processing, computer graphics and little to no use of computer vision. Techniques in 3D computer animation involves mainly computer graphics and computer animation. In contrast, problem of converting an existing 2D cartoon video into 3D will mainly require computer vision techniques like 3D scene reconstruction, tracking and pose estimation. Computer Vision community mainly uses real videos captured using one/more cameras. So far only Bregler et. al. [5] and Gleicher et. al. [25] use cartoon videos for motion processing.

Bregler et. al. in [5] presented a technique called “Cartoon Capture and Retargeting” which can be used to capture motion from traditionally animated cartoons and retarget it onto 3-D models, 2-D drawings, and photographs. Users have to manually identify keyframes in both input and output media, as well as corresponding parts between each keyframes. The system do not recover out-of-plane motion from the input, instead 3D information is inferred during designing of keyframes for the 3D model. For example if in input images the hand of a character points away from the viewer (into the image plane) then the keyframe designer should interpret it and design keyframes accordingly. This process puts lot of burden on the keyframe designer, more complex is the motion of a character more number of keyframes are required.

Gleicher et. al. in [25] injects some of the expressiveness of traditional animation into existing motion capture 3D animation, by incorporating few drawings created by a skilled animator. First, user of the system manually adjusts the skeleton of a 3D model according to drawings, then user has to specify correspondence between the drawings and the 3D model, finally system automatically warps 3D model to look like 2D image and these changes are propagated to the nearby frames.

Our work is different from these two in type of information inferred from images - while these techniques do not recover 3D from images we want to do it. Due to lack of literature in computer vision using cartoon videos remainder of this section will discuss techniques for reconstructing 3D from images and capturing articulated motion (for example humans) from monocular images.
2.1 3D Modeling Techniques

Humans have an uncanny ability to perceive and analyze the structure of the 3D world from the visual input. Humans operate effortlessly and often have little idea what the mechanisms of visual perceptions are. But with computers, analyzing 3D structure from 2D images is a fairly involved problem. There exists a rich pool of algorithms [24, 29] for deriving a 3D scene description from one or more 2D images, but there exist no robust and general solution which can handle all types of problem instances, although practical structure from motion methods do exists. Some techniques for inferring 3D structure from the images are discussed below.

1. **Photometric Stereo** [36] uses three images obtained using light sources from three different directions. Both the camera and the objects in the scene are required to remain stationary. By knowing the surface reflectance properties of the objects in the scene, the local surface orientation at points illuminated by all the three light sources can be computed. One of the important advantages of this method is that points in the three images are perfectly registered since the cameras and scene both are static. The primary disadvantage of this method is that it is an indirect method and it may be not be practical to employ an imaging system in which the illumination is so carefully controlled.

2. **Shape from Texture** [29] techniques exploit the image plane variations in the texture properties such as density, size and orientation. The texture gradient, defined as magnitude and direction of the maximum change in the primitive size of the texture elements, can determine the orientation of the surface.

3. **Shape from Focus** algorithms [29] exploits infinite depth of field of the optical systems, due to which only objects which are at a proper distance appear focused in the image whereas those at other depths are blurred in proportion to their distances. These algorithms model images as the convolution of focused images with a point spread function determined by the camera parameters and the distance of the object from the camera. The depth is recovered by estimating the amount of blur in the image and using the known or estimated line spread function.

4. **Motion Parallax** [29, 24] have been used to reconstruct 3D the from sequence of 2D images. When images in stationary scene are acquired by a moving camera, the displacement of the image plane coordinates of a scene point from one frame to another depends on the distance of the scene points from the camera. Thus points nearer to the camera move faster then points away from the camera, this information can be used to find relative depth of the points from a sequence of images. This is also true for the moving object and a stationary camera.
5. **Shape from Contour** [29] is a method for construction of a 3D model of the object based on a sequence of images of the object, where the object’s contour is the only interesting feature in the image. These techniques try to emulate human visual perceptions, which can infer object shapes from shape of the 2D boundaries in the image. For example given an ellipse in the image, immediate 3D interpretations are disk or sphere. If shading or texture is uniform a disk will be favored, if the shading or texture changes appropriately towards the boundary, the sphere will be favored.

6. **Structure From Motion** [29] is one of the most powerful technique, which can estimate relative or absolute depth of static or moving objects from a sequence of images obtained from a single or multiple cameras. It is based on the principle, “Given three distinct orthographic projections of four non-coplanar points in a rigid configuration, the structure and motion compatible with the three views are uniquely determined up to a reflection about the image plane”.

7. **Shape from Shading** [16] techniques infer 3D using information about the variation of shading in a sequence of images. These techniques work on principles like smooth object becomes increasingly darker as the surface normal becomes perpendicular to rays of illumination; a planar surface have a homogenous appearance in the image with the intensity proportional to the angle made between the normal to the plane and rays of illumination. In addition to constraints imposed by radiometric principles, shape from shading assumes that surfaces are smooth in order to calculate the surface orientation parameters.

The shape recovered from techniques discussed can be expressed in one of the several ways: depth $Z(x, y)$, surface normal $(n_x, n_y, n_z)$, surface gradient $(p, q)$, and surface slant $\phi$ and tilt $\theta$. The depth can be considered either as the relative distance from camera to surface points, or the relative surface height above the x-y plane. The surface normal is the orientation of a vector perpendicular to the tangent plane on the object surface. The surface gradient, $(p, q) = (\partial z/\partial x, \partial z/\partial y)$, is the rate of change of depth in the x and y directions. The surface slant $\phi$ and tilt $\theta$, are related to the surface normal as $(n_x, n_y; n_z) = (l \sin(\phi) \cos(\theta); l \sin(\phi) \sin(\theta); l \cos(\phi))$, where $l$ is the magnitude of the surface normal.

### 2.1.1 Challenges

Cartoons often do not have shading (they are flat shaded) or texture, therefore techniques like shape from shading and shape from texture cannot be used. Inferring 3D from a single camera video is possible using structure from motion techniques, however it requires scene to be projected through physical laws, e.g., perspective projection. Cartoons are hand drawn therefore projected model is not clear and may change from frame to frame. Also the character may undergo non-rigid deformations through frames, these differences in
character shape must be taken into consideration. These difficulties, make cartoon videos a challenging application for inferring 3D from a sequence of images using vision based techniques. An important feature of cartoons is that they have well defined contours as they are hand drawn, in fact most of the cartoons do have a well defined black contours. In this research we propose to use graphics based freeform object modeling techniques to overcome difficulties in cartoon videos. Next section discuss a freeform object modeling tool ‘Teddy’.

2.1.2 Teddy

There exists freeform design tools, which allow users to create a 3D object interactively through a series of strokes using a mouse. One of the recent freeform design tool for creating 3D objects is ‘Teddy’ proposed by Takeo Igarashi [17]. Teddy is a sketching interface for quickly and easily designing freeform models such as stuffed toys. In teddy user draw several 2d freeform strokes specifying silhouette of the object and the system automatically generates a 3D polygonal mesh model of the object. This is done by making the wide areas fat and narrow areas thin. Advantage of using such a system is user doesn’t have to manipulate the control points.

2.2 3D Motion Capture Techniques

In last two decades motion capture from video have gained lot of attention from computer vision researchers due to wide range of applications - movies, games, surveillance, control areas, advertisements, movement analysis and animations. Gleicher discusses potential of vision based motion capture for creating animation [13] and challenges of motion capture from animation point of view [14]. For more detailed treatment refer to surveys on motion analysis [12], [2] and vision based motion capture [27].

The process of motion capture from video can be divided into three parts - initialization, tracking and pose-estimation. Initialization refers to finding camera parameters like focal length, pixel aspect ratio, position and orientation; adaptation to scene characteristics; and model initialization. Tracking is the main step in the process of 3D motion capture which can be defined as establishing coherent relations of the subject and/or limbs between frames, coherent in this context means clear and consistent, emphasizing that subject/limbs must be consistently identified from frame to frame. Pose-estimation refers to the process of identifying how a subject and/or individual limbs are configured in a given scene. For the purpose of 3d motion capture, pose estimation can be seen as a post-processing step in a tracking algorithm or it can be active part of the tracking process.
For the purpose of review of 3D motion capture techniques several criteria can be used to classify techniques -

- parameter estimation method - single or multiple hypothesis.
- image features used - edge, intensity, silhouette etc.
- model based or model free.
- representation used to model body parts - cylinders, quadrics, cones, deformable etc (3D models); points, blobs, box, silhouettes (2D models).
- kinematic representation used - euler angles, twists, quaternion etc.
- 2D vs 3D approaches.
- number of cameras used - single or multiple.

In this work we classify motion capture from video based on information inferred during tracking from the images - 2D or 3D. A subject can be tracked to infer either 2d position and/or orientation of individual limbs using points, blobs or boxes or by employing a 3d model build by cones, cylinders, quadrics, or deformable objects 3d configuration of limbs can be obtained. In the case when tracking is done to infer 2d information from images, pose-estimation for the purpose of 3D motion capture becomes a post-processing step of tracking. Thus 3D motion capture can be classified into two categories - 2D tracking followed by pose estimation and 3D tracking.

Rest of this section is organized into 4 parts - first part discuss difficulties in 3D motion capture from monocular images, second part discuss techniques for tracking articulated structures in 2D, third part gives the overview of pose estimation and fourth part details techniques for 3D tracking.

2.2.1 Challenges

There are number of difficulties in motion capture from monocular video. Some of them are discussed here:

1. **Depth 3D-2D Projection Ambiguity**: Projecting a 3D scene into images suppresses the depth information which is a fundamental problem in the computer vision. A simple example of depth ambiguity can be seen in figure 2, where a wireframe model of a cube can be interpreted in more than one way. For the case of 3D human motion captured from monocular images depth ambiguity can be observed in terms of forward and backward flipping of the limbs with respect to the image plane. Solving for depth
ambiguity is an ill-posed problem, which have multiple solutions.

2. **High Dimensional Representation**: For articulated objects we are interested in for example humans or animals, the model must have large number of parameters (minimum 30 for human) in order to reproduce reasonable class of motion with some accuracy. In such a high dimensional space, estimation is extremely computationally demanding and exhaustive random search is practically infeasible. The key problem is efficient localization of good cost minima (likelihood peaks).

3. **Physical Constraints**: Apart from being able to represent articulated object accurately model chosen must also respect physical constraints. In particular, they have to ensure that body parts do not penetrate each other and that the joints only have physically valid interval of variation. Addition of constraints make the search space lot smaller, thus many false solution can be pruned. But handling constraints is not so easy for continuous optimization based methods. Apart from checking which constraints are active, one also needs to decide where to look during next search. Simply rejecting solutions not respecting constraints makes the solver too slow.

4. **Self-Occlusions**: For the highly articulated structures we are interested in self-occlusions are very frequent. For reliably tracking such a high DOF articulated object there is a need to accurately predict self-occlusions. But the problem is during self-occlusion some of the parts cannot be observed in the image thus during model-to-image matching step the model parts corresponding to the occluded parts in the image can be matched incorrectly with some ‘other’ body part or even to something similar in the background. Thus there is a need for constructing a likelihood surface that realistically reflects the probabilities of different configurations under occlusion.

5. **Observation Ambiguities**: These ambiguities arise when certain model parameters cannot be inferred from the current image observations and depends on the image features used. For example when limb is straight and edge based features is used then rotation of limb about its own axis cannot be recovered. But this would not be a problem with intensity based cost function.
6. **Background Clutter**: This is another common vision problem, where background contain many features that resemble body parts, for example pillars. The problem is not exactly due to presence of such features in the background but it is due to our limited current ability to construct models which can reliably extract high level structures from images.

### 2.2.2 2D Tracking

Cham et. al. [21] proposed a multiple hypothesis tracker which uses 2D kinematic model called Scaled Prismatic Model (SPM), introduced in [20]. Each link in SPM describes the image plane appearance of an associated body part. Each SPM link can rotate and translate in the image plane. The link rotates at its joint center about an axis perpendicular to the image plane modeling articulation, and translation models the distance between two joint centers of the adjacent links capturing foreshortening that occurs when 3D links rotate into and out of the image plane. They use condensation [19] style tracker along with non-linear local optimization for registering SPM with images over time. But unlike condensation which uses non-parametric representation for multi-modal distribution they use piecewise gaussian representation. This representation is based on the assumption that underlying pdf have well defined modes, and that these modes capture the essential structure of the pdf which is required for accurate tracking.

Black et. al. [23] presents a techniques for tracking and recognizing human motion by modeling human body as a set of connected planar patches - called *cardboard person model*. Each limb in the human body is approximated as a planar image patch and tracking is achieved by using parameterized motion model [4] of optical flow for each patch along with the connectivity constraints between the adjacent patches.

Guo et. al. [15] presents an approach for tracking and motion recognition using 2D stick model having 10 sticks and 6 joints, assuming static background, motion parallel to image plane and orthographic projection. They used background subtraction to extract human silhouette from images, then compute skeleton of the silhouette and construct a gaussian potential field around the skeleton. Finally correspondence between stick model and image silhouette is established by searching the potential field for stick figure with minimal energy. Though authors reported good results, keeping motion parallel to the image plane constrains them to use simple motions like walking and running.
2.2.3 Pose Estimation

Recovering 3D pose from the tracked joint location in 2D images is an ill-posed problem. For known limb lengths there is forward/backward flipping ambiguity for each limb, leading to $2^{#\text{limbs}}$ inverse kinematics solutions. In order to overcome these ambiguities some other information has to be used. Some of used information is - 3D motion clip similar to motion being captured, joint angle constraints, some limbs parallel to image plane etc. Even with introduction of constraints problem still remains underconstraint and for current techniques some form of manual input is required.

Cham et. al. [10] presents a method for computing the 3D motion of articulated models from 2D joint locations in the images. They estimate the pose of the model at each frame by minimizing the distance between projection of a joint in 3D and manually marked 2D joint location. Since this formulation is under-constraint they introduce number of constraints in the system - joint angle constraints, to ensure physically valid pose; dynamics, to ensure smooth continuous motion; and 3D key frames, despite adding constraints system is still underconstraint so they use partial or full 3D posed for some keyframes. The distance is minimized subject to these constraint using a Gauss-Newton least-squares method.

Park at. al [28] uses motion library for resolving depth ambiguity in recovering 3D configuration from given 2D joint locations. Assumption is user must know what motion to reconstruct beforehand and thus have a similar motion clip. Given 2D locations of joints and a reference 3D motion clip, system is initialized by interactively marking key-times (moments of interaction between actor and surroundings) in images as well as motion clip. Using this information system first warps reference motion to establish the time correspondence with the motion in the video, and then reconstruct a motion by deforming the timewarped reference motion guided by the features in the video.

Kakadiaris and Barron [3] use anthropometric data from [1] to construct an eigen space of possible Stick Models (SM). User have to manually mark bones whose orientation is almost parallel to the image plane, in addition to projection of joints of the SM. System is initialized by estimating approximate limb lengths using the eigen space and initial estimate for pose using almost parallel segments. Then using a hierarchical non-linear solver anthropometric and pose data from a single image is obtained. The basic assumption used is that there are number of bones whose orientation is almost parallel to the image plane.

Remondino and Roditakis [31] presents a technique which uses scaled orthographic camera model and human skeleton with known relative lengths of limbs to recover 3D pose of humans from an image or video. Such a formulation has a limitation that only relative
depths between two end points of a segment can be recovered, thus user has to resolve this ambiguity. Since this framework only gives relative distances (thus adjacent frames can have very different locations in 3D), to use it in sequence of images simple translation motion model is udrf for the joints.

2.2.4 3D Tracking

Park et. al. [22] presents a model-based singularity free automatic initialization approach to capture human motion from static background monocular video sequences. They used simplified 3D model with each limb modeled as a cylinder, head as sphere, and all joints have 1 DOF. They considered motion parallel to image plane. System extracted human silhouette from background (using background subtraction), and approximate it with a polygon. Model parameters are optimized for maximum overlap of projected model silhouette with extracted image silhouette.

Bregler and Malik [7], [6] demonstrated a technique for extracting 3D motion from a video using a 3D articulated model approximated by ellipsoidal blobs. Tracking is achieved by using a region based tracker, where shape of a region is represented by a support map. Their main contribution is development of a motion model for a 3D kinematic chain under scaled orthographic projection using twists of exponential maps, to obtain linear system of equations. Though authors report good results no information is provided about the accuracy of the method, nor for what class of postures and human body dimensions does the method work.

Sminchisescu and Telea [33] presents a method for human pose estimation using a 3D articulated model covered by ‘flesh’ of super-quadratic ellipsoids. Their method work by first extracting silhouette of the human from the image and then estimating parameters (pose) of 3d model by non-linear optimization methods. Parameter estimation is based on two components, first component maximizes area of overlap between projected model silhouette and image silhouette; second term pushes model inside image silhouette such that contour of projected model matched as close as possible to contour of image silhouette. Distance level-set, computed using Fast Marching Method [32], is used to speed up computation for second term.

Deutscher et. al. [8] presented an approach for motion capture using a 30 DOF 3D model fleshed with conic section with elliptical cross-section. For image measurements they used edges and silhouette. The tracking is achieved by using the Annealed Particle Filter (APF), a probabilistic density propagation method related to condensation [19], which avoids local minima with the help of simulated annealing using noisy dynamics. Another
purpose of simulated annealing process in this paper is being able to track using lesser particles in comparison to condensation which might be helpful for real time applications. Authors have extended AFP to automatically partition search space [9] based on variance of parameters being optimized. Though reported results are very good, the test sequences are obtained by 3 cameras and using a constant background.

Rehg and Kanade [30] proposed an approach to the local tracking of articulated objects based on a layered template representation of self-occlusion. Using a kinematic model, templates are ordered according to their visibility to the camera and represented using an direct graph called occlusion graph. From occlusion graph visibility order of the templates can be predicted for the next frame and tracking can be achieved by registering overlapping templates with an image through gradient-based minimization of an SSD residual.
3 The Proposed System

The aim of this work is to research on the development of a set of tools that can be used for converting an existing video footage of a 2D cartoon film into a 3D computer animation, where the viewer has the choice of changing the viewpoint. Converting the entire video with all the fine details kept will be a computationally and economically infeasible task. The idea is to use the original story and sound tracks, model the main characters while replacing the irrelevant details with enhanced visual appearance and new background without affecting the original storyline. In this work, four tools necessary for converting the video footage into a 3D computer animation are identified:

- 3D Object Modeling Tool
- 3D Motion Capture Tool
- Facial Expression Modeling Tool
- User Interface

In this research we will be working on 3D object modeling and motion capture tool, along with their user interfaces. Facial animation is essential for a full animation system, but including it will make scope of this work too extensive. In the following pages, an approach for 3D modeling and motion capture which we will be following is discussed.

3.1 3D Object Modeling Tool

The 3D object modeling tool is responsible for obtaining 3D model of the objects occurring in the input video. We classify the objects needed to be converted into three types.

1. main characters like Tom, Jerry or Popeye. These characters must be modeled in detail since they will be the center of interest for most of the time.

2. side characters for example a man walking in the background or a character playing with a dog. Such objects can be represented using a generic library of the objects without affecting the storyline. In the example of dog, even if we replace a dog with another type of dog it would not affect the story. These objects need not be as detailed as main characters.

3. inanimate objects like a sign board or a table. These objects do not require great detail and for most of the purposes a standard library can be used for such objects.

We propose 3D models of the main characters in the input video can be obtained in the three ways - digitize from available hard toys of characters using a laser scanner, create
by hand using software’s like 3D Studio Max or Maya (since target users for the proposed
system is animation studios, and they are good in modeling), and using a semi automatic
computer vision system. So far we have used teddy [17] to create 3D models from manually
marked contours, figure 3 shows an example. The created model is good for the process of
tracking and pose estimation but for final animation a more detailed model is required. We
propose to combine vision based structure from motion technique with teddy to obtain a
more detailed model of the characters in the video.

3.2 3D Motion Capture Tool

Once the 3D model of a desirable object from the video is obtained, the next step is to
capture its motion from the video. Capturing 3D motion from a single video for the articu-
lated objects is an inherently difficult problem as discussed in the literature survey. While
modern tracking techniques are increasingly robust and can track multiple hypotheses [19]
[34] [8] they are still limited in the range of motions that can be reliably tracked - certainly
no tracking method works all of the time.

So far we have been evaluating standard computer vision approach to capture 3D mo-
tion from a video involving automatic tracking of joints followed by pose-estimation, with
manual initialization of the first frame. The problem of 3D motion capture from single
video is inherently difficult due to the singularity of the video input especially when dealing
with articulated motion. Solutions of such kind of problems require some modelling or prior
knowledge of the motion. In our cartoon scenario, these auxiliary inputs will be incorpo-
rated to achieve our goal of creating a high quality 3D cartoons from traditional cartoons.
As cartoon scenarios are varied and unconventional limited only by the imagination of the
artists, we will need a semi-automated approach to deal with the various exaggerations that
we see in traditional cartoons.
4 Results

We have conducted some experiments on modeling, pose estimation given the 2d correspondences and tracking. This section describe our algorithms.

4.1 Modeling

For creating model of a character in the cartoon video we use a free form 3d object modeling system called ‘Teddy’ [17]. The algorithm is implemented in C++ and is used in Maya animation package as a plugin. User first selects a frame in the video which have frontal view of a character to be modeled. The selected image is loaded in Maya and user is asked to draw contours on the image corresponding to character in the Maya with the help of a spline tool. These contours are sampled to create closed polygons and 3D model is created by using Teddy. Figure 3(c) shows an 3D model of Aladdin created using image shown in Figure 3(a), some of the curves marked by the user are shown in 3(b). Squares represent control vertices of splines.

4.2 Pose Estimation

We have developed a system for estimating 3D pose of an articulated model from the given 2D correspondences. User have to manually specify 3D pose for the first frame and for rest of the frames 3d pose is computed automatically. Our idea is based on very simple observation - given the projection length of a line in the image plane and known length of the 3D line there are only 2 possible solutions for the 3D pose, one out of the image plane and one in the image plane, figure 4 illustrates an example. When projected lengths of
the limbs are known, the problem of estimating 3D pose is reduced to choosing one of the
solution for each limb in the articulated model. The pose of a limb is determined by first
predicting the position of a joint in the next frame using a constant velocity motion model
and then choosing a solution which is closer to the predicted position, assuming motion
between two frames is not large, see algorithm 1 for more details.

\[ \cos(\theta) = \frac{l}{L} \]

Figure 4: Two possible solutions for projection of a limb with known length

We have tested our idea on simulated datasets obtained from web in the form of BVH
files which contain description of articulated model and motion parameters for all the frames.
2D correspondences from 3D motions is obtained by projecting joints on the image plane
using orthographic projection. Figure 5 shows some of the frames of 350 frame sequence of
a person somersaulting. Dark color lines represents projection of the estimated pose of the
limbs and light colored lines are projection of limbs in the input 3d motion.

```
1: struct Joints
2: {
3:     vector3d 3dposition
4:     vector2d projection
5:     Joints *parent
6: }
```
Algorithm 1 Pose Estimation Algorithm

1: for all joint ∈ Joints do
2:     parent = Parent(joint)
3:     limblength = \| parent → 3dPosition_t − joint → 3dPosition_t \|
4:     projectedlength = \| parent → projection_{t+1} − joint → projection_{t+1} \|
5:     ZDistance = \| limblength − projectedlength \|
6:     solution1 = (joint → projection_t.x, joint → projection_t.y, ZDistance)
7:     solution1 = (joint → projection_t.x, joint → projection_t.y, −ZDistance)
8:     if (predictedPosition − solution1) < (predictedPosition − solution2) then
9:         joint → 3dPosition_{t+1} = solution1
10:     else
11:         joint → 3dPosition_{t+1} = solution2
12:     end if
13:     joint → velocity_{t+1} = α * joint → velocity_t + (1 − α) * (joint → 3dPosition_{t+1} − joint → 3dPosition_t)
14: end for

Figure 5: Pose estimation results for human performing somersault
4.3 Tracking

One of the most popular approach to tracking is the condensation algorithm [19], however it is well known that condensation algorithm cannot be used for high degree of freedom articulated objects since it will need huge number of samples to approximate posterior density. It has been shown by MacCormick and Blake [26] that number of particles needed for fair representation of posterior density is \( N > \left( \frac{D_{\text{min}}}{\alpha d} \right) \), where \( d \) is number of dimensions (\( d \approx 30 \) for humans). Survival diagnostics \( D_{\text{min}} \) and particle survival rate \( \alpha \) are both constants with \( \alpha \ll 1 \). Clearly when \( d \) is large normal particle filtering becomes infeasible. There are two ways to overcome the difficulty of the condensation - optimize each particle to find near by minima in cost function or partition the search space, we used the later approach.

In order to partition the search space we represent 3D kinematic model as a graphical model with each joint as a hidden node and limbs as the edges in the graph. The sequence of images to track then becomes observed data for the graphical model. The problem of tracking with this representation can be formulated as estimating belief from the graphical model. Our tracker is based on the recent works [35] [18] on non-parametric belief propagation. We used 3D kinematic model with 17 joint and total of 30 DOF, each node in the graphical model have 6 variables - 3 position and 3 orientation.

The framework described in [35] [18] assume that state of a hidden variable can be deterministically computed given the state of its neighboring variable. But in the case of 3D kinematic model, given the position and orientation of a joint only position of its child joint can be estimated; and given the position and orientation of a joint no information can be inferred about its parent joint except restricting its position on a sphere. Thus we cannot use their message update algorithm directly, so instead of taking product of messages using a gibbs sampler we combined all the messages into one and then draw a sample based on the weight to approximate the message product. Another difficulty for using belief propagation is to obtain observations, since we cannot observe limbs separately in the images we cannot obtain measurements for each node separately. To overcome the difficulty of measuring the state of a joint \( i \) for a set of particles, we draw a sample for the remaining joints based on their current marginal distribution. Now for each particle of joint \( i \) we use same pose for rest of body and obtain measurements by constructing a full body pose using state of joint \( i \) from the current particle.

We have tested our algorithm on synthetic examples generated by projecting a 3D human mesh model using opengl. Figure 6 shows two examples.
Figure 6: Tracking Results
5 Conclusions

Motivated by the success of 3D computer animation and popularity of the traditional animation we propose to modernize traditional animation with new looks and feel while preserving original story and actions, so that we can enjoy traditional animations once again. In this work we propose to develop a set of tools that can help to automate this process. We have identified four tools necessary for conversion of the traditional 2D cartoons into 3D, these are - 3D object modeling tool, 3D motion capture tool, facial expression animation tool and user interface. In this research we will be working on 3D object modeling and motion capture tool, along with their user interfaces. Facial animation is essential for a full animation system, but including it will make scope of this work too extensive.

After having some experience with vision based motion capture techniques and considering that cartoons scenarios are varied and unconventional limited only by the imagination of the artists, we propose a semi-automated approach to deal with the various exaggerations that we see in traditional cartoons. In this report we have discussed some of the techniques relevant for converting the traditional cartoons into 3D and some of the challenges for reconstruction and motion capture process that makes cartoon videos a particularly challenging application for the vision based techniques. We also presented some promising results obtained for tracking and pose-estimation process.
References


