Master Thesis

Czech
Technical
University
in Prague

F3

Faculty of Electrical Engineering
Department of Control Engineering

Autocompletion algorithm for simple trajectories

An algorithm that will automatically complete a simple demonstrated trajectory

TAN Wei Xin

Supervisor: Mgr. Radoslav Škoviera, Ph.D.
Supervisor–specialist: Mgr. Karla Štěpánová, Ph.D.
Field of study: Cybernetics and Robotics
Subfield: Cybernetics and Robotics
May 2019

Acknowledgements

First and foremost, I would like to thank my supervisors, Mgr. Radoslav ’koviera, Ph.D. and Mgr. Karla ’t¥pánová, Ph.D., for providing me with the opportunity to pursue this topic. Without their prior work on the Asus XTION PRO 3D sensor, the HTC Vive virtual reality system and the trajectory dataset, this thesis would have not gotten to the level it is at right now. Their guidance has also been important in allowing me to finish this piece of work in a timely manner.

Besides that, I would also like to thank Markus '¯¯ ' Stroot for his ingenuity and his suggestions related to the field of computer vision without him this thesis may have ended up stuck forever on the image processing part. Many thanks also goes Felix ȍò ' Staudigl for introducing me to the wonders of feature selection and for his infectious optimism which has kept me going even when all hope seemed lost. On top of that is Flavio ' ' Kreiliger who I have to thank for being one of the best workmates I've ever had as well as Jasna Petric for always being a fun person to work with.

I would also like to thank Š “ for leaving her comfort zone and staying by me for so long as well as for being willing to support me however she can. Finally, last but not least, many thanks go to my family for placing so much of their trust in me. Without their advice, love and support, I would have never made it as far as I have today.

Declaration

I declare that the work presented in this thesis was developed independently and that I have listed all relevant sources of information used within according to the guidelines on observing ethical principles in preparation of university theses.

Prague, 23. May 2019

. . . .

TAN Wei Xin

Abstract

Motion planning is one of the core problems that is being studied extensively by robotics researchers in the present day. Among the many techniques available, planning via automatic path/trajectory generation is one of the most widely used approaches. This thesis has implemented a system using tools from computer vision, computer graphics and supervised machine learning which can 'autocomplete' a user demonstrated trajectory segment on differently shaped blocks. This means that the final trajectory generated will be based on the shape of the block with the user's demonstration superimposed on it. The aim is for the trajectory to be utilized in planning the motions of an industrial robot. In the process of developing this system, this thesis provides a comprehensive review of the subject fields utilized and covers the basic intuition behind the algorithms used in the system.

The final results of this thesis show that it is possible to automatically generate smooth and continuous trajectories that are non-photorealistic using information from a human made trajectory segment. Although the system is functional, it should be considered as a proof of concept rather than as an industrial level implementation. There is much improvement to be made to this thesis' system before it can be considered fit enough to be deployed in an industrial setting.

Keywords: trajecory generation, digital image processing, shape recognition, computer graphics, non-photorealistic rendering, machine learning, trajectory classification

Supervisor: Mgr. Radoslav ’koviera, Ph.D.
B-611a
Jugoslavskych partyzanu 3
160 00 Praha 6

Contents

Figures

Tables

Chapter 1

Introduction

In the modern era, people have become increasingly reliant on automation and robotics which means that there is a need for these machines to have a wide variety of capabilities at their disposal. The number of human robot interactions have also been on the rise which means the design of future robotic systems will have to be more accommodating towards humans. In addition, there is also a need for robots that can perform highly monotonous work such as applying glue with a glue gun in order to free up people's time to tackle more difficult tasks.

1.1 Motivations

The planning of robot motion is a very large field. It has many existing solutions[LaV06] and yet, is still filled with challenges especially in an industrial setting[KC]. The classical approach to motion planning for industrial robots is designing joint position, velocity or even torque profiles using programming by demonstration (more well known as imitation learning)[BCDS08]. A human instructor moves the industrial robot in a desired manner and stores the corresponding actuator information thus 'teaching' the robot in a kinesthetic manner. This approach is especially ideal for repetitive tasks. If some sort of visual system for object recognition is included, the robot can behave flexibly under many different conditions.

The downside of imitation learning is that it is usually difficult to scale scale a task and/or transfer it to a different robotic setup. This means that every single case must be considered and that the human instructor must guide the robot through the entire process perfectly. Hence, human instructors have to be experts at the task being taught. Furthermore, imitation learning also requires them to come into physical contact with the robot. This type of human-robot interaction is not without its risks[KC05]. As such, all of this makes it more difficult for the robot to learn new processes. Due to these issues, industry has begun to utilize a number of different technologies such as virtual[FR99] and augmented[DGL08] reality. This is expected to reduce physical human-robot interaction as well as to reduce the reliance on expert human instructors.

Since the user is not expected to be an expert and human-robot contact should be minimized, there is a need for the robot to deduce the user's intentions. This means that not only should the robot determine what its operator would like it to do but also, that the robot should filter any 'distorted' commands. This is due to the fact that non-expert users should only be able to imagine a task in being performed in an 'ideal' manner. They should not have the skill to replicate their own thoughts exactly. This ability to 'autocomplete' a human operator's intentions would make an industrial robot more adaptive and useful since it would require less training and supervision.

1.2 Goals

This thesis aims to propose a system which will 'autocomplete' trajectories based on a human demonstrated one. This will allow an industrial robot with a glue gun attached to its end-effector to use these trajectories for path planning without the need for physical human-robot interaction. More specifically, with the use of a 3D sensor and a virtual reality system, the goals of this thesis are the following.

1. Detect the shape of a target using the 3D sensor
2. Classify the type of trajectories recorded by the virtual reality system
3. Determine the 'ideal'/non-distorted version of the trajectory classified 2.
4. Generate a new trajectory that contains the trajectory found 3. superimposed over another trajectory based on the shape detected 1. in

In the case of 2., the trajectories should be identified without needing the user to 'complete' it. This means that drawing any pattern along the path of a circle should only require one segment of the pattern for the system to 'autocomplete' the rest of trajectory. On top of that, the pattern need not be perfect 3. should filter out any distortions e.g. a wave pattern with inconsistent periods and amplitudes should become a sine wave with constant periods and amplitudes.

1.3 Contribution of thesis

The main contribution of this thesis is a method for procedurally generating trajectories that contains complex curves superimposed on another simpler curve. The complex one is based on user demonstration while the simpler one is based on shape detection. This includes:

1. a computer vision algorithm that preprocesses visual as well as depth images and determines the shape/edges of the target from them
2. a supervised machine learning algorithm that preprocesses 3-dimensional trajectory data consisting of xyz coordinates and classifies them according to type

3. a computer graphics algorithm that generates 'ideal' versions of trajectories using mathematical equations and superimposes them onto another trajectory

This thesis is structured as follows. Chapter 1 lists the motivations for this thesis' topic and what this thesis hopes to achieve. Chapter 2 explores the relevant work in the areas that this thesis employs. Chapter 3 elaborates upon the methods used by this thesis to achieve its goals. Chapter 4 goes on to present the results of implemented system on the dataset collected for this thesis. Chapter 5 examines in more detail the system and results of this thesis. Chapter 6 analyzes the aspects of this thesis that could be developed further. Appendix A provides the pseudo-code that made this thesis' results possible. Appendix B contains a glossary of the acronyms and symbols used while Appendix C lists the references utilized in this thesis.

Chapter 2

Related Work

The work presented in this thesis is closely related to 3 distinct subfields. These are vision based trajectory generation, non-photorealistic rendering (NPR) and trajectory classification. All three have been researched extensively and there have been a number of works on NPR robotics [LMPD15][SSGG19]. These works have utilized ideas from all 3 subfields although mainly for painting/picture reproduction purposes. This thesis, on the other hand, is more focused on path/trajectory planning in an industrial setting. As such, this thesis draws upon many previously used approaches and integrates them to develop the final system.

2.1 Vision based trajectory generation

In robotics, trajectories are smooth functions of time which specify the robot's and/or its end-effector's position [LP17]. Very often, trajectories are designed to satisfy constraints such as positional waypoints or workspace/velocity/acceleration/torque limits. Trajectory generation also has to take into account obstacle avoidance. Since trajectories should be smooth, they are usually approximated by polynomial functions.

For useful trajectories to be generated, at least part of a robot's workspace must usually be known. This information can be obtained in many ways. One of the most common is through the use of vision systems. In the case of [ANSL03], their system was developed to automate the cutting of embroidered material. Using a digital camera equipped with an ambient light filter and a laser diode, they were able to detect a difference in light intensity where the seam of the embroidery occurred. They used this information to generate a trajectory which they smoothed using either a moving average filter or a Recursive least square filter. Their approach, however, was very specific to their task. The edges their system were detecting were of non-negligible thickness which is why their use of a laser diode was effective. Also, their camera was mounted close to the end-effector of the robot unlike in this thesis' case. The vision system used in this thesis was stationary.

Another related work is [CVPS12] where the system was developed for spray painting. In their case, a barrier sensor was used to obtain the edge and surface primitives/lines of the object (the algorithm behind this process was not elaborated on). Then, an Eulerian trajectory was determined so that every primitive/line was visited once. Although their approach is particularly useful for generating trajectories with no overlap, it presumes that the object edges are already well known. Also, image processing was not discussed which is not the case for this thesis.

[MPW14] is another work of interest. They utilize image edge detection to generate a trajectory for applying sealant to engine blocks. Here, the authors use a single digital camera and apply Canny edge detection to the images that they acquire. Using integral projection functions, they then proceed to determine the engine block's outermost edge. After that, they filter it to remove jumps/skips between pixels before using it as a trajectory. This thesis uses a similar methodology for shape recognition although there was not as much control over experimental area's lighting.

Finally, there is [SL17] which was developed for drawing pictures using a differential drive robot. They also use Canny edge detection as the basis of their trajectory. They go further by approximating these edges with cubic B-splines which results in continuous joint positions, velocities, accelerations and even jerk. Although their work did not utilize other image processing methods besides Canny edge detection, their use of B-splines demonstrated its usefulness for robot trajectory generation.

2.2 Non-photorealistic rendering (NPR)

In traditional computer graphics, the aim of rendering is usually photorealism i.e. the reproduction of an image or scene such that it resembles the output of an actual camera. NPR, on the other hand, is more subjective in that it aims to create an artistic representation [SS02]. Although NPR encompasses many techniques ranging from classical graphics rendering to sketch-based modelling, this thesis will be primarily exploring the statistical and mathematical modelling approaches to curve synthesis — the creation of output curves that are similar but contain information that is not present in their inputs.

The multiresolution approach to curve synthesis is used by [FS94], where the idea of decomposing curves into multiple levels of detail using wavelets was introduced. [BSS07] improved and extended it further to cover different sketch styles. Curve decomposition is done by using two matrices known as analysis filters. One downsamples/coarsens the input curve while the other preserves the features removed by the previous matrix. Curve synthesis is then performed by utilizing another 2 matrices known as synthesis filters. They reverse the effects of the analysis filters. This process is done iteratively and generates what the authors call a 'filter bank'. The 'bank' contains filter matrices for every level of detail. Even though this method was considered, it was determined to be more complicated than necessary for the purpose of this thesis. Nevertheless, it is possible that their approach may be quicker for longer and/or more complicated curves. Further studies will have to be done to ascertain this.

Another approach to curve synthesis has been explored by [HOCS02]. The authors also treat curves as decomposable into multiple levels of details like [FS94]. Feature extraction and reconstruction, however, is performed using local transformations. This synthesis of curve analogies is done by first finding the rigid transformation between curve A (base) and curve B (input). This transformation is then applied to curve A' (style) and the result is split into multiple segments. A cost function, $E(B')$, that characterizes the difference between:

- the shape of A' and B',
- the shape relationship of A to A' and that of B to B',
- the relative position and orientation relationship of A to A' and that of B to B',

Figure 2.1: Example of curve results from [HOCS02], $A : B :: A' : B'$

is then defined. For every segment, a rigid transformation around the center of mass of the B' segment that minimizes $E(B')$ is computed. This thesis' approach is similar and has the advantage of allowing for patterns that had no overlap unlike in that seen in Figure 2.1.

There is also a statistical approach to curve synthesis as seen in the works of [KMM$^+$02]. The authors use a Markov random field (MRF) to synthesize curves based on several user provided example curves. Another one is [SD04] where Hidden Markov Models (HMM) are used to synthesize appropriate curves onto new illustrations. In the case of [KMM$^+$02], user provided curves are stored as a spline and vectors perpendicular to the spline. These are used to characterize the curve's features. The MRF is then used to map similarities of a curve segment to the probability of that segment being added to the new, synthesized curve. For [SD04], their HMM is trained on multiple sets of detailed and non-detailed 'control' curves. This is to determine the correspondence between them. From that, detailed curves are then synthesized from non-detailed user drawn curves. The key to their approaches' success is the use of large training datasets which this thesis did not have. Also, the method used in this thesis only requires the user to demonstrate part of the desired trajectory. The system being developed by this thesis is not aiming to replicate the user's input completely but to generate an 'ideal'/non-distorted version of it. As such, statistical modelling was used not in the curve synthesis portion of the thesis. Instead, it was used in the trajectory classification portion which is elaborated in Section 3.4.1.

2.3 Trajectory classification

Much of the work on trajectory classification is centered around 2-dimensional cases. 2D trajectories are usually categorized based on spatial and temporal features i.e. position, velocity etc. Many of those approaches, however, can be extended to the 3-dimensional case. Though in 3D, the importance and relevance of the 2D features need to be re-evaluated. Also, new features need to be determined in 3D to aid classification. This is what happens in [BKS06] where the authors explore classifying 3D trajectories created from the Australian Sign Language (ASL). By developing affine-invariant features from trajectory coordinates, the authors were able to use HMMs to recognize ASL words not oriented in the same manner i.e. the signs for the same word could be rotated or stretched. These new features are the centroid distance function (CDF) and curvature scale space (CSS). Both were adopted by this thesis in order to improve classification results. Unlike [BKS06] though, the trajectory datasets collected for this thesis were constrained to the same area of the coordinate space since they were all performed on the surface of the targets. This meant that the xyz coordinates could be used together with the CDF and CSS for training the classifier used in this thesis.

TraClass [HLG08], on the other hand, uses region and movement pattern clustering to tacklet the 2D trajectory classification problem. This framework allowed the authors to approach trajectory classification in a hierarchical manner. It sorted the detected features and gave preference to higher level ones for classification. The procedure behind TraClass can be described in 4 general steps

1. Each trajectory is split into line segments based on their change in direction using the minimum description length principle [GMP05]. Segments from similar trajectories as well as those from similar classes end up grouped together.

2. The groupings are used to identify regional clusters via a grid structure. A regional cluster is one where there are relatively many trajectories of one class in comparison to other classes. This is done recursively with 1. to ensure as many clusters as possible are found.

3. The groupings are also utilized to establish movement pattern clusters within a user-specified neighborhood. Each group must contain segments from the same class otherwise they are treated as noise. This is also done recursively with 1. to ensure as many clusters as possible are found.

4. Both clusters based on region and on movement pattern are merged for class identification. Regional clusters are given higher priority compared to movement pattern clusters.

TraClass' approach was effective at performing classification on 'top-down' trajectory data. Nevertheless, it was not suitable for this thesis' dataset as most of the 3D trajectories recorded did not have significant regional differences. Movement pattern based clustering may have been suitable but seemed unlikely. This is because TraClass' experiments were shown to work well on trajectories generated by different sources (e.g. deer vs cow) whereas the trajectories for this thesis were performed by one subject multiple times.

Another work dealing with trajectory classification is [PMF08]. The use of Support Vector Machines (SVM) for abnormal trajectory detection is examined by the authors. Here, they introduce an improved approach to tuning SVM parameters. SVMs have been previously used for trajectory classification and is also one of the classifiers used by this thesis. The authors' method of selecting optimal parameters, however, is more applicable for outlier detection. It is less relevant for this thesis compared to multiclass identification as there were more than 2 types of trajectories present in this thesis' dataset.

Similar to [BKS06] is the work of [FD09]. It addresses the classification of hand movement and orientation. In order to label analogous motions correctly, the authors of [FD09] did not take into consideration temporal information of the trajectory (otherwise fast and slow motions would be classified differently). They also normalized trajectory length so that it would not affect classification. Also, they smoothed and divided the trajectory into discrete parts. Then, they use spherical and cylindrical coordinates to compute orientation features. After building a learning table from those features, they then utilize it to develop a classifier based on Bayes' theorem. They also use Shannon entropy to further improve results. Although similar preprocessing was performed (such as data smoothing and resampling), it was determined that their method of calculating orientation features was not appropriate for this thesis. This is because trajectory coordinates in the dataset corresponded to the tip of a glue gun instead of the position and orientation of a human hand. Moreover, this thesis' dataset came with its own orientation profiles but it was determined that they varied too much to be useful for training the final classifier.