5-4-2023

AN INCREMENTAL BASED APPROACH FOR 3D MULTI-ANGLE POINT CLOUD STITCHING USING ICP AND KNN

Pankti K. Patel
Rowan University

AN INCREMENTAL BASED APPROACH FOR 3D MULTI-ANGLE POINT CLOUD STITCHING USING ICP AND KNN

by

Pankti K. Patel

A Thesis

Submitted to the
Department of Electrical and Computer Engineering
College of Engineering
In partial fulfillment of the requirement
For the degree of
Master of Science in Electrical and Computer Engineering
At
Rowan University
May 2023

Thesis Chair: Ying (Gina) Tang, Ph.D., Professor, Department of Electrical & Computer Engineering

Committee Members:
Li, Jie, Ph.D., Assistant Professor, Department of Electrical & Computer Engineering
Zhu Cheng, Ph.D., Assistant Professor, Department of Civil & Environmental Engineering

Dedication

I want to dedicate this manuscript to God, my family, and my advisor Dr Tang.

Acknowledgements

I would like to show my appreciation to my advisor Dr Ying (Gina) Tang. She has guided and supported me throughout these years and always motivated me to fulfil my dream. The skills and knowledge that I have gained are the things that will stay with me throughout my life. I look forward to facing new challenges that come my way knowing that I am prepared to take them on.

I would also like to thank my family, my fiancé and my friend for their love, support, and motivation throughout this journey.

Abstract

Pankti K Patel
AN INCREMENTAL BASED APPROACH FOR 3D MULTI-ANGLE POINT CLOUD STITCHING USING ICP AND KNN
2022-2023
Ying (Gina) Tang, Ph.D.
Master of Science in Electrical and Computer Engineering

The basic principle of stitching is joining or merging any two materials or objects. 3D point cloud stitching is basically stitching two 3D point cloud together. 3D point cloud stitching is an emerging topic and there are multiple ways to achieve it. There are various methods for stitching which all have changes throughout the time. The existing methods do have shortcomings and have ignored the multiangle stitching of a same model or an object. This shortfall leads to many deficiencies in the ability of a stitching algorithm to maintain accuracy over the period.

In this work I have introduced a new approach for an iterative based approach for 3d multi-angle point cloud stitching using ICP (Iterative closest point algorithm) and KNN (K-nearest neighbor). The design follows an incremental approach to achieve the results. This is a novel approach of stitching multiple 3D point clouds taken from multiple angles of a single bust. The framework is evaluated based on the stitching results provided by the algorithm capability of stitching multiple point cloud into a solid model.

Table of Contents

Table of Contents (Continued)

List of Figures

List of Tables

Chapter 1

Introduction

3D models and its application have widely grown over the years. The first ever 3D model was developed in 1960. Back in the time 3D modelling was a type of fascination because it played a major role in feeling realistic and realism. It was the first time when just like a real object the 3D models were able to move, rotate, manipulate, invert and played around in all the possible methods on the screen. Every 3D model consists of a 3D mesh and generating a 3D mesh is known as 3D modelling. 3D modeling has made its mark in all the industries. Creating a virtual three-dimensional model of a physical object using software is called 3D modeling. Some examples of 3D modeling software are Blender, Maya, Sketup, AutoCAD, Revit and many more. In the current era there is not even a single industry which does not use 3D models in their development.3D models are widely used in animations, games, artificial intelligence, construction, movies, architecture, medical, virtual world, and its applications are countless. Future of 3D models is much more advance than today. The future holds many advantages and applications of 3D models in 3D printing and virtual reality.

3D models can be divided in mainly three categories of models like wire frames, surface model and solid model. Wire frame models are the model which is defined by defining each edge of the physical model (wired mesh) where two smooth surfaces intersect or by connecting the vertices and arch of the object. Calculating and rendering of a wire frame model is simple and easy so they are mainly used for complex 3D models and real time systems which includes exterior topologies. It is used early in the production process to establish the basic outline of the structure of a page before any content and designing is developed. The second category of 3D model is Surface models. Surface models are very similar to the paper models and give the best realistic view of an object. They are very easy to play around and modify according to the requirement. Surface models have an upper hand over wire frame models as it helps to define the edges of the object along with its surfaces using a polygonal mesh. 3D surface models are extensively used in the automobile industry to build car bodies, engines, and turbine blades. It also helps to validate imperfection in models, apply smoothness and examine procedural surfaces. The third category is Solid model which involves geometric / mathematical modeling and computer graphics. The main objective of solid models is to maintain physical accuracy. These are widely used for engineering prototypes as it involves weight, density, center of gravity and stress. 3D solid modeling is also used in computer graphics, medical practice, testing, in 3D virtual visualization and experimental science research. All the three models differ from each other in terms of complexity, visualization, and accuracy.

In former times, 3D models were only used in mathematical modeling and data analysis.3D modeling is defined as the development of an object by using mathematical representations of an object in three dimensions. 3D models have become an integral part of many industries like gaming, architecture, entertainment, animation, interior designing, scientific / medical imaging and many more. In olden days medical schools widely used the 3D models made from wax, bronze in the teaching of medicine but today they have evolved into 3D and started using 3D modeling approaches in teaching, research, diagnosis, and treatment. One of the widely used 3D modeling application is 3D printing. It is a manufacturing process of building a physical object by imposing thin layers of material in progressive manner by referencing a digital 3d model.

In recent years, advancements in computer simulations and virtual environments have created an ever-increasing need for methods to scan real objects and reconstruct them into accurate three-dimensional (3D) models. Generating 3D models using legacy modeling software is time consuming, complicated, and expensive [31]. In virtual environments, reconstructing real objects significantly reduces the time required to create realistic 3D objects. Additionally, accurate scans of real objects also open new possibilities for computer aided or entirely digitized methods for applications such as skin health diagnoses and manufacturing defect recognition. However, in all applications, easy and rapid object scanning is vital, as is the accuracy of the resulting 3D model [1].

1.1 3D Point Cloud

The principal cause of point clouds is generating 3D designs or models. In simple terms we can explain point cloud as a group of data markers in space. These group of points represents a 3D object or a model with X, Y, Z coordinates.3D point clouds are basically a visualization of the millions, or trillions of points which come together to form a model or an object. There are multiple ways of creating 3D points like 3D laser scanners, LIDARS technology and techniques. Point clouds are becoming more applicable in wide range applications as well multiple industries. Point clouds are known for their availability, accuracy and their density.3D point cloud has multiple applications like CAD models, Construction, metrology, animation, rendering and medical purposes. Alignment of point cloud depends on the other corresponding points and this procedure is called point set registration. Point set registration method is also called point cloud registration or scan matching. It is the technique of finding the best combination of spatial transformations to align two-point clouds. The Spatial transformations include scaling, rotations, and translations. Point clouds can be stored in various formats like PLY, PTX, OBJ, XYZ, PCD, DOT AND LAS. ASCII file format is the most common file format of point clouds.

1.2 Motivation of Thesis

Computer vision, image processing and reconstruction have evolved drastically in these years. There are a wide range of applications where these technologies are used. Generating 3D objects widely requires algorithms in this virtual world as the virtual world and artificial intelligence applications are at its peak. Artificial intelligence applications are so powerful that they adapt the behavior and the actions of the user input and predict the future inputs.

Motivated by the recent and existing methods of automatic point cloud stitching and recent advancements in machine learning methods, this study presents an efficient algorithm for stitching concurrent point clouds into single 3D models. Specifically, our proposed algorithm augments the iterative closest point (ICP) stitching algorithm with k-nearest-neighbors (kNN) clustering. Using this combined approach, our method iteratively minimizes the error between two neighboring point clouds, locating the corresponding points on both point clouds to combine and create a solid 3D model. Compared to a standard ICP algorithm, the addition of kNN allows us to obtain a more accurate matching between two neighboring point clouds.

1.3 Objective of Thesis

The primary objective of this study is to obtain a 3D model from multiple point clouds taken from different angles. To achieve this goal many methods and algorithms were explored and investigated in terms of effectiveness, robustness, efficiency, and accuracy. A combination of two algorithms is designed and implemented to achieve the goal of generating a solid bust or model from multiple point clouds. The combination of algorithms used are iterative closest point and K-nearest neighbor. Furthermore, various statistical and error-based analysis were carried out to measure the performance of an incremental based approach for 3d multi-angle point cloud stitching using ICP and KNN.

The 3D point cloud stitching is a technique to take input as point clouds and merge them together. Traditionally the algorithm starts by taking the input which are point cloud of a solid bust from different angles all around the object. All the collected data is then provided to the algorithm for the future process which includes, down sampling, denoising, registering, aligning, and merging them together to receive a final sold bust out of multiple point cloud data. By following this approach, the output is received as a solid model with few anomalies.

1.4 Organization of Thesis

This study comprises of five chapters, as follows:

Chapter 2 is basically a literature review on the methods which I have used in the algorithm along with the existing techniques as well as the issues associated with it. The five-subtopic named incremental stitching, Iterative closest point algorithm, K nearest neighbor, various methods of stitching and problems of the existing methods. The literature review and analysis are done on those different approaches of stitching and algorithms used in the study. A brief discussion is reviewed and analyzed on the iterative closest point algorithm and K nearest neighbor. Incremental stitching also plays a vital role in this dissertation and is evidently explained in this thesis. There is a detailed study described on the present approaches which are available in the market for stitching. Further to better understand the algorithm there is a review on the existing methods and their various problems which exist on the models which we use for the study. This chapter will provide a detailed explanation to the readers and researchers on the machine learning algorithms used for incremental based approach of stitching.

Chapter 3 gives a brief overview of the stitching algorithm used in the study. It includes four subtopics named elements of the proposed algorithm, role, and detailed explanation of the iterative closest point algorithm and K-nearest neighbor algorithm. This chapter explains the architecture and flow of the proposed algorithm and provides the pseudo code for the method. The pseudo code depicts each step of the proposed algorithm and explains how the data is being processed and finally stitches multiple point clouds into a model.

Chapter 4 will help the readers visualize the different results of various inputs provided to the algorithm. It includes five subtopics named implementation of the different input data like bust and sculpture, stitching with different variants, stitching results with ICP and Stitching results with ICP and KNN. The readers will also be able to see the different implementation results using various variants like implementing incremental approach and increasing and decreasing the number of input data. This study also gives a comparison between the results of the algorithm with only iterative closest point algorithm and the results of the modified iterative closest point algorithm i.e., iterative closest point algorithm with K nearest neighbor.

Chapter 5 summarizes the experimental finding of the proposed algorithm and literature review conducted on the existing methods available. Furthermore, this chapter acknowledges the contribution and the limitation of this approach along with the future scope of this study.

Chapter 2

Background

The research began with the fascination about the 3D point clouds and evolution of 3D models.3D modeling has become a very common approach for visualization. 3D modeling is known for creating a 3D object using the simulated software, the objects could be simple shapes to complex high-polygon models. The idea behind the research is to develop an efficient algorithm which can help the multiple 3D point clouds to be stitched together and create a model.

The benefit of this research will help in creating a 3D model which is the key in visualization, 3D modeling, Artificial intelligence and many more other evolving applications.

2.1 Iterative Closest Point Algorithm

The iterative closest point algorithm is mainly used for minimizing or reducing the disparities between two-point clouds. It is frequently utilized to reconstruct 2D and 3D models. In ICP, it is named as source and target for the input provided to the algorithm. Either source or the target one is always kept fixed and then the corresponding pairs are found. For the target point, it finds the best matching point in the source and this process is continued until all the points in the point cloud or image have corresponding matching point in the source. The corresponding pairs are the combination of rotation and translation matrix calculated based on the source and target points. There are plenty of variants which can be tweaked according to the needs of the problem. The standardized theoretical approach for the iterative closest point cloud algorithm is

1. For all the points in the reference point cloud a matching pair is obtained in the target point cloud.
2. Estimation of the matching pair is calculated. Matching pair compromise rotation and translation. This step will also include rejecting the outliers.
3. Transforms the target point cloud based on the obtained transformations.
4. Repeat the steps and iterate over all the point clouds.

Mathematical approach for the iterative closest point algorithm is explained below. The problem statement is to align point clouds. The first step is to always find a feasible transformation which will allow to align the two-point clouds with the lowest error function also in the same coordinate systems.

Given the two pointclouds P and Q

Corresponding relationships are established between two-point clouds P and Q. Each point in the point cloud P finds the corresponding matching point in point cloud Q.

R and T denote the rotation and transformation(translation) matrix.

For an arbitrary 3D point cloud set P ,3 × 3 rotation matrices. The 3-dimensional point set N the input is denoted by $(x1, y1, z1\boxed{x1, y1, z1})$ and the output is denoted by $(x2, y2, z2\boxed{x2, y2, z2})$.

$$[X_2\ Y_2\ Z_2\ ]\boxed{\begin{bmatrix}\mathcal{X}_2\\ \mathcal{Y}_2\\ \mathcal{Z}_2\end{bmatrix}} = \mathrm{R}\ [X_1\ Y_1\ Z_1\ ]\boxed{\begin{bmatrix}\mathcal{X}_1\\ \mathcal{Y}_1\\ \mathcal{Z}_1\end{bmatrix}} + \mathrm{t}$$

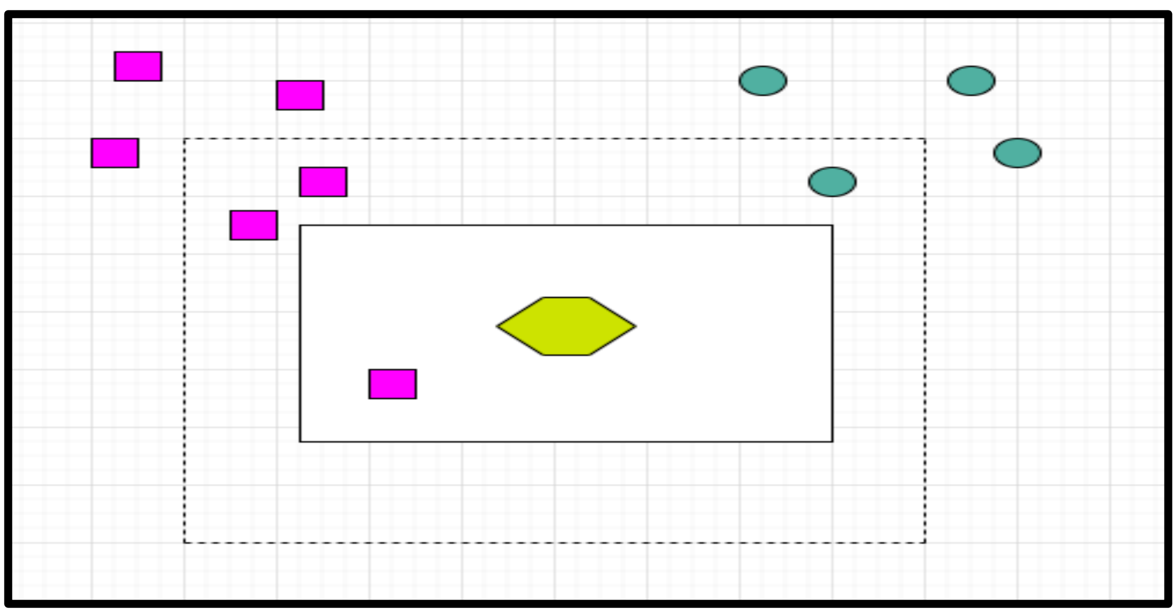
The Iterative closest point algorithm is mainly based on the optical matching method that is least square method. The algorithm iterates itself for selecting the corresponding matching pairs and calculates the rigid body transformation until it reaches convergence.

2.2 K Nearest Neighbor (KNN)

It is one of the methods that is used for classification and regression. The function is approximated locally, and all calculations are delayed until function is evaluated. This algorithm is dependent on distance so by normalizing the training sample in the data set will highly contribute to increasing its performance. An optimizing strategy for classification and regression is to allocate the weights to the offering neighbors so that the contribution of the closest neighbors is the height then the farther neighbors.

Figure 1

A Simple Illustration of the K Nearest Neighbor Algorithm



The above figure displays the simplest example of the KNN classification algorithm. The training data is the beige green hexagon. It can be classified by either squares or circles. If K=4 the hexagon will be classified as pink squares as there are three squares compared to one circle. If K=10, the hexagon will again be classified as a pink square as there are six squares compared to four circles. Classification will be based on the value of K so choosing the right value of K is very much important. There are several characteristics which should be kept in mind while deciding upon the value of K. Firstly when the value of K is less or K=1 the algorithm and the predictions becomes unstable as averaging is not possible. Contrarily when we increase the value of K, projections become more stable due to averaging and selecting the mode as the value of K. But this will retain a certain value of K if we push the value of K far beyond, we will start having errors. So, pushing far beyond a certain point for the value of K is not advisable. When averaging and picking the mode of the dataset as the value of K, picking an odd number is advisable to avoid going into loops so it can act as a loop breaker.

2.3 Various Methods of Stitching

Multiple stitching algorithms are available and are well researched based on the set of requirements as well as the data available. Each algorithm has a specific purpose to fulfill and will expect a particular set of input data. Stitching itself compromise three significant areas which need to be handled and they are feature detection, feature registration, and alignment.

One of the most common methods of 3D object scanning is using 3D laser scanners, which can accurately capture multiple angles of the same object, allowing a user to easily reconstruct an accurate 3D model. However, laser scanners are economically prohibitive and unwieldy, often requiring multiple people to successfully operate and scan objects. More cost-effective and portable methods of 3D object scanning can be found in visual scanning using structured light methods and depth cameras. However, visual and depth camera methods do not automatically combine scanned objects into complete and solid 3D models. Instead, these methods return discrete point clouds from each angle captured on a target object [1].

It is necessary to stitch multiple point clouds to retrieve solid 3D models from discrete, segmented point clouds. Although this process can be performed manually, it is extremely time-consuming, and automatic methods for point cloud stitching would greatly improve the usability of both visual and depth camera-based scanning methods. Several methods [21][6] to stitch point clouds exist in the literature, but there are several issues with the existing methods: 1) methods require modified or transformed input data, adding additional complexity and time to the algorithm, and 2) methods are prone to high error or inaccuracy in stitching methods. As such, there is still room for fast and accurate methods for creating a 3D model from a series of concurrent point clouds [1].

A common approach for aligning neighboring point clouds is the iterative closest point (ICP) algorithm [1]. The functional step is to determine the transformation that best matches the point clouds with a given correspondence [22]. ICP implicitly assumes that there is a good overlap between the source and target point clouds so that stitching can easily converge [21][4]. While simple and effective, ICP presents the following practical shortcomings owing to the alignment assumption: (1) high computational cost [23]; (2) it is prone to converge to local minima [23]; and (3) it is easily affected by outliers [24]. To address these limitations, researchers have devoted efforts to modifying the standard ICP algorithm, leading to several ICP variants in a range of applications in medical fields [19], remote sensing [12], autonomous driving [12][25], robotics [26], and aviation [27].

Feature detection involves identifying the content of the data set and checking for certain properties or the requirements. Entire algorithm depends on the correspondences found in this step. Determining the robust matching pair is very crucial to get best stitching results with minimized errors. Automated Point Cloud Correspondence Detection for Underwater Mapping Using AUVs [8] is developed to find the best matching pairs/correspondences using a sequence of steps which include image generation, feature extraction, image matching and false match rejection. Feature can be anything from the data set may it be edges, points, or an object. Feature Registration comprises of identifying the overlapping features or points of the two concurrent data. The execution of this step depends on the number of features selected. If there are a huge number of features to be matched it will be time consuming and affect the accuracy of the alignment. Alignment is the step where the actual alignment of the registered pairs takes place. The overlapping features should be aligned in such a way that they minimized the overall distance and overlapping pixels.

Among the modified approaches, one of the simpler yet effective modifications of ICP is to weigh point pairs to modify their impact on the transformation or to reject unmatched pairs outright [28]. Another method of Hybrid ICP [29] is proposed which optimizes the data association method and error metric based on the live image of an object and the current ICP estimate [29]. An alternative method to improve the registration of ICP is by introducing a point-to-plane metric that utilizes the surface information of the point sets to improve registration accuracy [24]. Another line id based on identified features. For example, Zong et al. applied an improved scale-invariant feature transform (SIFT) method to help match points [30]. However, as in [2][10], this method requires data manipulation to convert the 3D data into 2D images. Once the point cloud is converted, the 2D images are stitched together before being converted back. In this approach, the conversion to and from 2D has a high computational cost, and information such as the normal directions and colors of points can be lost in the conversion. Furthermore, this conversion may not always be possible because of different data formats [10]. Therefore, methods that operate directly on the 3D point cloud are desirable to avoid the aforementioned issues [1].

Parallax-tolerant Image Stitching Based on Mesh Optimization [9] is a large parallax image stitching method which combines 2D homogrpahy based on mesh optimization. Loose homogrpahy and mesh-based wrap is used to align the images. Once the alignment is completed a graph cut algorithm is applied to complete the final stitching output. A tree structure mosaic model is used to stitch the sequences of input images. The proposed method was designed to overcome ghosting and severe distortion while overlapping feature points.

Automated high-speed stitching for large 3Dmicroscopic images [7] is a simple, fast, and efficient method to stitch large, microscopic images by determining the optimal adjacency of tiles using minimum spanning tree. It follows a two stage stitching approach first is pair-wise registration followed by group registration. Pairwise registration is the method which helps calculate the translation and distance score between all the respective tiles pairs of the images. Group registration is graphical where each tile is considered as node and the distance between two tiles is selected as the edge weight between two tiles nodes. This will help information of the graph and we can calculate the minimum spanning tree which helps to find the order of tiles.

Another significant study which was carried out was using Least Square Conformal Maps (LSCMs). LCSMS can help in establishing 2D recurrent parametric domains for 3D surfaces which will help in simplifying the 3D stitching problem to a 2D stitching problem [10]. Diffeomorphism is a key attribute of LCMS which helps in detecting and removing the duplicated regions in the original 3D surface. This existing algorithm has simplified the 3D stitching problem by converting into 2D by connecting the neighbor's in 2D recurrent parametric domain. This proposed algorithm is robust to occlusion, noise, and different resolutions of the input [10].

TeraStitcher - A tool for fast automatic 3D-stitching of teravoxel-sized microscopy images [11]. Tetrastitcher has an algorithm which can stitch teravoxel-sized tiled microscopy images in less time and resources. It basically matches the specific features and data points from the images which make the tool efficient and can also perform on large images which have different characteristics. Algorithm use the microscopy technique which is leading to teravoxel-sized tiled 3D images at high resolution, thus increasing the dimension of the stitching problem of at least two orders of magnitude. This tool also successfully tested on mouse brain images with micrometer resolution. Data Samples collected by algorithm are illuminated by a thin sheet of light and scan the axis perpendicular to the illumination plane. Objects are then translated along the scanning axis, so obtaining a tile as a stack of 2D slices. Combining this technique with a special procedure to clear tissue makes it possible to acquire large specimens, such as brain images.

3D surface reconstruction and panorama stitching based on LCD-based calibration and multi-baseline stereo matching [14] used LCD (Liquid Crystal Display) based calibration and multi-baseline stereo matching algorithm. In the calibration method it used a 2D control field composed by the flat Liquid Crystal Display (LCD), on which lots of points(circle) with known coordinates are displayed. The stitching algorithm is a combination of 7 step processes including drawing 2D control field on the LCD, LCD based calibration, Rotation shooting by hand at actual survey site, multi-baseline stereo matching, Accuracy evaluation, 3D surfaces generation and Panorama stitching.

2.4 Problems of Existing Methods

Various existing methods are available in the market and researchers have done their research and have multiple applications. The existing methods application depends on its input, its applications, and its requirements. Existing methods are efficient but have their own limitations.

Several methods convert the 3D point clouds or 3D data sets into 2D images and then stitch them together and convert the final stitched image into a 3D point cloud or 3D data set. This type of method is very much efficient as it helps avoiding projection errors, cheaper for large datasets as well as easy to amend. But it has disadvantages like conversion of 3D to 2D and vice versa becomes very much expensive, time consuming as well as error matrix increases. Conversion is not always possible in certain scenarios due to the type of input used or the manipulation which is required for certain sets of input [10]. This approach is also very time taking as it will add additional steps of conversion at the start as well as at the end.

Medical industry is also a huge contributor in stitching algorithms. Medicine application and studies have really improved and are at its peak of innovations. Several applications of image processing are developed for the medical industry. There are several methods available to stitch 3D microscopic images. Handling microscopic images can get a little difficult as they are huge in size. Stitching images can be handled by pairwise registration as well as group wise registration. Medical data sets are huge in size ranging from hundreds of gigabytes so sometimes it gets difficult to fit the large datasets into a panoramic view [7]. Microscopic images have multi color channels so it becomes difficult to build a mechanism which can choose the best signal to noise ratio channel. So, in most of the applications the color channel is eyeballed for the reference of stitching [7].

Memory usage or space usage have been a major concern in data processing or image processing. Optimizing the algorithm is the main concern while developing new algorithms for image manipulation or 3D processing. There are many algorithms which fail to address the issue of space usage. The existing software solutions do not seem adequate to address the additional requirements arising from these datasets, such as the minimization of memory usage and the need to process just a small portion of data [11]. Fast and efficient processing can be achieved by increasing the hardware capabilities that is by increasing the number of processors of the tool used for processing. But this is not always doable due to cost constraints. So, it is always better to make the algorithm more efficient than relying on the hardware solutions.

Parallax is also considered one of the challenging aspects to address. Parallax is the displacement of the original position of an object to the viewing position when views through different line of sights. It is measured by using the slope of the angle between the line of different viewing position. Due to linear perspective, closer objects have greater parallax compared to distant objects as they have smaller parallax. Image stitching with large parallax has long been an important and challenging issue in computer graph and vision [9]. Parallax stitching with mesh optimization is only applicable to images and not applicable to 3D models.

LCD based calibration algorithm can get the high precision camera parameters, And multi-baseline stereo matching, but image distortion rate is high in the close-range images using LCD. Use of LCD has the limitation to use points cloud generation in urban areas with multi-baseline stereo matching [14].

Feature Detection is identifying the details from an image or a point cloud. The stilting results depend on the common features identified between the source and target data. An algorithm is developed [8] but its limitation is to sonar multibeam data. This method also follows the path of image generation from existing point cloud. This step involves converting the point cloud into overlapping submaps and each map is converted into an image. The algorithm which this research is proposing eliminates this step and the raw point cloud as the input without manipulating it.

Chapter 3

The Stitching Algorithm

Motivated by the issue of automatic point cloud stitching and recent advancements in machine learning methods, this study presents an efficient algorithm for stitching concurrent point clouds into single 3D models. Specifically, our proposed algorithm augments the iterative closest point (ICP) stitching algorithm with k-nearest-neighbors (kNN) clustering [1]. Thus, with our proposed algorithm, we make the following contributions:

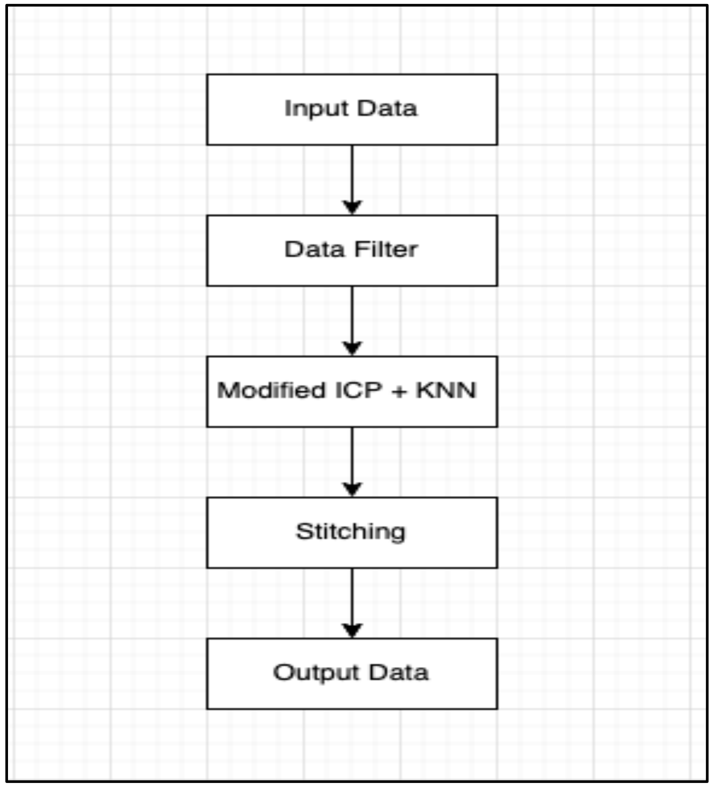
1. We propose a general-purpose iterative algorithm for creating a solid 3D model from multiple concurrent point clouds scanned from the same object.
2. Our proposed method augments an iterative closest point algorithm with k-nearest neighbors, creating a method that achieves high efficiency and reduced error compared to standard algorithms.
3. The proposed method operates directly on 3D point clouds, avoiding computationally expensive transformations.

3.1 Elements of Algorithm

The Stitching Algorithm comprised multiple steps to reach the final stitching model.

Figure 2

Workflow of Stitching Algorithm



3.1.1 Data

Collecting data is an important aspect of the algorithm. Selecting the object model with differentiating features is very important as features are the essential part for reconstruction. To capture the data, we used object models, RealSense™ depth camera and brown GUI. We are using intel real sense depth camera and its SDK cross platform library which helps in calibration and in providing the intrinsic and extrinsic calibration information. The algorithm uses three different solid models made from POP (Plaster of Paris) with a lot of differentiating features.

The position of the camera and the object model is also important as we are not calibrating the camera in each iterative step, so the placements of the instruments also play a vital role in data collection. We place the camera and the object in the same horizontal axis having their center of origin as fixed for both the camera and object. Please refer to Figure 3. The camera is calibrated and stays at a fixed location throughout the collection of data. We collect a series of pointclouds of the object model by rotating the object at an angle keeping the center of origin fixed. We place the object on a flat surface having the same shared axis of the camera. We then start displacing the object at a certain angle and keep on iterating through the displacement process 360 degrees until the original position is achieved. We have used clockwise and anti-clockwise direction movements to capture various sets of the data. Algorithms have captured numerous data by manipulating the angles and total number of point clouds. Data collection is based on three major manipulations criteria like displacing the object at a greater angle, total number of point clouds in a set (starting from the origin and rotating the object for 360 degrees) and the depth of the camera to the object. We have tried various combinations of data sets to reach the most efficient combinations of data sets.

3.1.2 Data Filtering

The algorithm utilizes two methods to filter the data that is Down sampling and Denoising. The point clouds contain many points, so it is very much advisable to perform down sampling [13] of the point cloud before analyzing it further. Point clouds are ungrouped data points which define the shape of the object. Consuming point clouds directly without achieving uniformity of points can be challenging due to the unevenness and huge number of points. By reducing the size and grouping of the point cloud will help in making the further analysis of the point cloud efficient and optimized. This algorithm uses a box grid filter to down sample the data. The grid filter takes a grid size which will be the numeric value and creates a box around points. All the points within the box are merged into a single point and the colors and normal of all the points are averaged to calculate the colors and normal of the single point. The box grid filter helps in preserving the shape of the point cloud(object) and has proved more efficient than the random down sampling method. Algorithms have used various combinations of box grid filters to carry out experiments and record different results. We have achieved an ideal grid size which achieves the best results and reduces computational time.

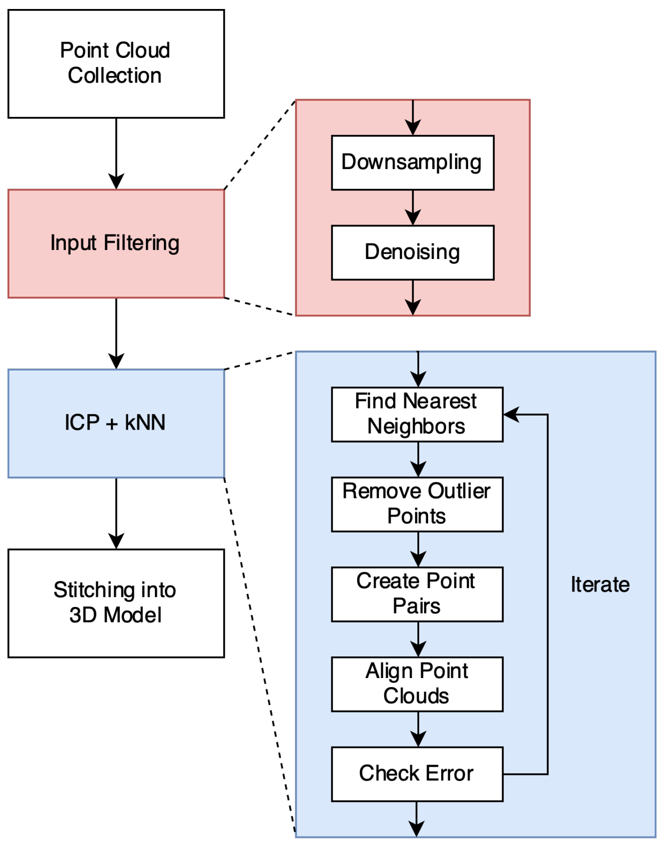
The point clouds are usually collected using cameras, 3D scanner which brings in a lot of noise and outliers. Before starting the registration, it is important to remove outliers from the raw data. The algorithm sets a standard threshold value (standard deviation) based on the mean of average distance of the point to its corresponding neighbor. It is very important to use an effective method which will remove the noise and outliers without losing the features of the 3D data. Down sampling and Denoising are both important steps for filtering the input data before performing registration.

3.1.3 Modified ICP and KNN

The standard methods are augmented to solve the problem statement. Fig. 1 gives a full flowchart of our proposed method. However, the proposed approach makes some assumptions. First, it assumes that there exists a set of paired points P that defines corresponding points in both point clouds. This assumption is valid when aligning two known or identical point clouds, as the paired points are either already known or easy to determine. With point clouds captured from real objects, the pairing is not known. The iterative closest point (ICP) approach is a means by which this pairing can be guessed to align two unknown point clouds.

Figure 3

Detailed Workflow of the Algorithm



ICP works through two basic steps: First, estimate paired points. This estimation is typically done through a similarity or distance metric. For example, for each point in B, the paired point is the nearest point in A. Using these pairs and a sub-optimal rotation and translation can then be computed and point cloud B can be shifted in the direction of point cloud A. Finally, the approach is iterated until some threshold criteria is reached, typically minimum error, maximum iterations, or minimum shift per iteration. Through this approach, two unknown point clouds can be aligned without knowledge of the paired points. However, ICP can still run into issues due to the lack of properly matched points such as slow convergence times, incorrect final transformations, or converging upon local minima instead of global minima.

To overcome the issues with the basic ICP algorithm this thesis proposes augmentation of iterative closest point algorithm and K nearest neighbors algorithm. Rather than estimating the single pair for all the points the proposed method finds the set of *k* candidate points. For each point in B, we first find the set of *k* candidate points. For each point in B, we first find the k nearest points in A using the Euclidean distance, shown in below Equation. We can then consider all k points in A as paired points to the single point in B and add k pairs to the set of paired points.

$$d_i = \sqrt{\sum_{j=1}^{n_B} (a_i - b_j)^2}$$

As this proposed method adds a much larger number of pairs to the paired list than standard, we can also remove any outlier pairs. To remove outliers, we discard any point pairs with distances above a certain threshold, typically to reduce the number of pairs to n_A.

Algorithm 1 Full Stitching Algorithm

Inputs: Source point cloud A

Target point cloud B

$minError$

1: **Initialize** $k = \sqrt{|A|}$

2: **While** $\varepsilon > minError$ **do:**

a. Call **Algorithm 2** to get list of paired points P

b. Compute t and R using Equation 1, and get total error ε

c. Transform $B = B * R + t$

3: **End while**

4: Transform $B = B * R + t$

5: Combine $X = A + B$

6: Apply a box filter on X, merging all nearby points into a single point, averaging color, and normal information

7: **Return** merged point cloud X

Algorithm 1 formalizes the entire process of stitching using two point clouds considering one as the source and one as the target. initially the optimal value of k is identified using the step 1. Step 2 is iterated over all the point clouds to find the matching pairs for each point in the source point cloud to target point cloud. *t* and R are calculated for all the pairs and total error ε is calculated. Using the translation and rotation the transformation is carried out for all points and lastly a box grid filter is applied to merge all the points colors and normal [1].

Algorithm 2 One Iteration of ICP with kNN

Inputs: Source point cloud A

Target point cloud B

Neighbor amount k

1: **Initialize** distance set $D = \emptyset$ and pair set $P = \emptyset$

2: **For** each point $b_i \in B$ **do:**

a. **For** each point $a_j \in A$ **do:**

i. Compute d_j from equation 2 as the distance between a_i and b_j

ii. Append $d_j \rightarrow D$

b. **End for**

c. Sort D by distance

d. **For** $a_i \in A$ corresponding to the k smallest values in D **do**:

i. Append $(a_i, b_j) \rightarrow P$

e. End for

3: End for

4: From P, remove highest-distance pairs until $|P| = |A|$ to remove outlier pairs.

5: **Return** P

Algorithm 2 showing a single iteration of augmented ICP with KNN. For each point in source point cloud a distance is calculated to the target point cloud. Distances are sorted and appended to the smallest k values. The points are filtered by removing the highest distances pairs until $|P| = |A|$ and later the outlies are removes. This sums up a single iteration of the algorithm [1].

The final step is stitching. The proposed algorithm uses a box grid filter to combine the overlapping regions. For each point in the region of overlap will be estimated to a single point based on the grid filter size. The colors and normal of the resulting point are averaged accordingly. Any undefined or infinite coordinates points are excluded from the merged point cloud. Formally, the stitching method uses the following steps.

1) Remove points with undefined or infinite coordinate values.

2) Concatenate arrays vertically.

3) Average the points in overlapping regions.

This approach not only makes the system more robust to noise, but also helps to prevent convergence on local minima since we use more than just the nearest point as a target location. Furthermore, the approach helps to speed up convergence since we consider not just the nearest point but k points, meaning that point cloud B is often shifted more toward the optimal location per iteration compared to a standard approach.

Chapter 4

Experiments

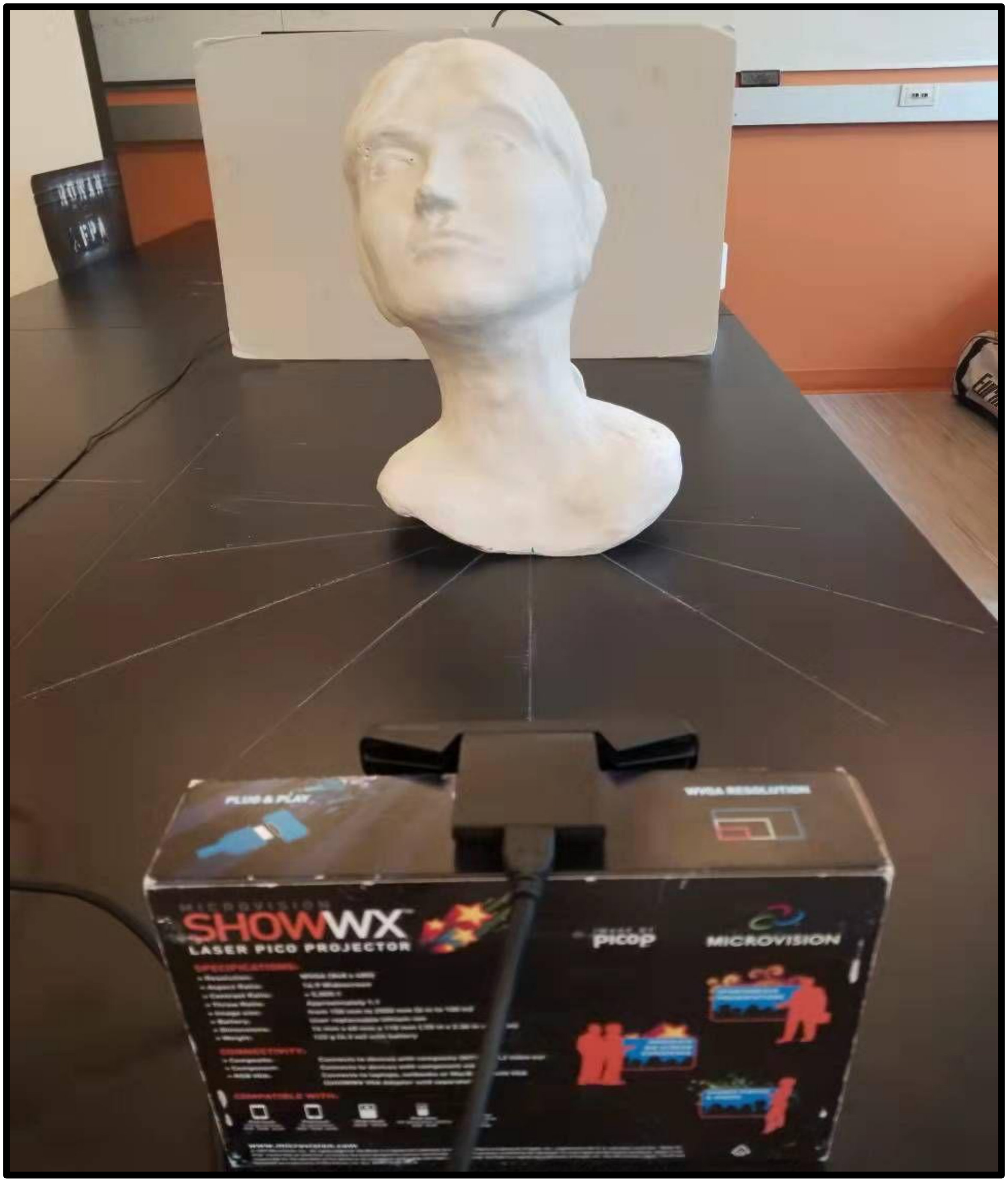
To explore the strengths and performance of the proposed algorithm, we carried out various experiments on the data collected. Our experimental setup incorporated an Intel RealSense F200 RGB-D camera placed on the same flat surface as the target object. The target objects are the sculptures made from plaster of Paris which are provided by the Art Department at Rowan University. Specifically, we utilized three different sculptures for testing shown in Fig 4.

Figure 4

Three Sculptures

Figure 5

Experimental Setup



The camera was kept still, while the object was gradually displaced following the marked angle lines underneath covering the entire object from front to back. shown in Fig. 5. The data were then captured and collected using our project-camera calibration system [18]. The stitching algorithm was implemented using MATLAB. Various combinations of tests were carried out to explore the optimal solution.

The few variations which showed evident difference in the stitching results are as follows:

1. Different types of target objects.
2. Varied depth of camera and object.
3. Varying the angle of displacement of the object.
4. Varied grid size parameter for down sampling.
5. Varying the size of *k.*
6. Varied box filter size for stitching.
7. Incremental approach.

The selection of the target object is the main and the first step. The algorithm is experimented with various kinds of objects which differ in terms of shape, size, surface, density, and material. The few objects with which we experimented were a cardboard box, a chips pack, a wood house and many more. For objects like cardboard boxes, we faced obstructions in reconstruction of the surface as the surfaces are flat. The observation from the experiments is that the algorithm works best for the target objects which have multiple distinguished features, and their surfaces are not flat.

By varying the angles between the object displacement there was an obvious difference in the results. The angle difference is correlated with the total number of point clouds. If the angle difference is small, we will have a higher total count of points clouds of an object covering 360 degrees which implicate sufficient overlaps between the source and the target point clouds. But if the angle differences are big and we have a smaller count of total point clouds of an object, the results will have anomalies as it will not get enough overlaps between the point clouds. The crucial aspect of this algorithm is to have the adequate number of common features between source and target to get better stitching results.

The important aspect of the algorithm is selecting a value for k. With too large of a value, the number of additional pairs are added to the pair list and it becomes very large, slowing convergence of the algorithm. As k approaches 1, however, the algorithm behaves similarly to standard ICP. Through our testing, we determined that $k = \sqrt{|A|}$, where $|A|$ is the number of points in A, provides a good balanced value for k with consistent performance throughout our testing.

Incremental stitching is a process of continuous stitching of the input data and keeps on incrementing the data through each iteration of the algorithm. It is a process of stitching a set of data in a cumulative manner over a particular range of data elements. It refers to the continuous and recurring fashion of stitching of input data elements and repeats itself till the n^{th} element of the data. I have followed two approaches of incremental stitching which are explained below. This first incremental approach of stitching, that it takes the first two inputs and performs all the operations and stitches them together and the output of the first two will be considered as the source for the third input. So, this incremental approach provides efficiency and robustness to the algorithm. The second incremental approach is that it follows a batch-based approach that is it takes the first ten-point cloud data and stitch them together and store its results, followed by the next ten-point cloud data and it continues to iterate over the point clouds data till the n^{th}amount of data. The result of each batch is then iterated over the algorithm and finally the merge or stitch point cloud is obtained. The batch number is decided based on the total number of point clouds. Batch based approach is proved to be more successful in terms of effectiveness, robustness and mean square error. The comparison of the two approaches leads us to the best output produced and the most efficient method which is batch based incremental stitching. The reason behind this is that it allows the data to be registered multiple times and data is aligned more effectively. This algorithm follows an incremental approach of stitching that is it takes the first two inputs and performs all the operations and stitches them together and the output of the first two will be considered as the source for the third input. Due to this incremental approach the algorithm can find the best correspondent matching pairs for registration and then the merging results are better. Overall, the incremental techniques have proved to be successful for 3d point cloud stitching as it provides efficiency and robustness to the algorithm.

In our case, we use two data filtering techniques: downsampling and denoising. Downsampling was applied through the pcdownsample function in MATLAB [20] and Denoising is performed using MATLAB's pcdenoise function [19]. In downsampling we use grid size to downsample the data. Manipulating the grid size did make changes in the stitching results as well as the total time taken to complete the reconstruction and stitching. By increasing the grid size, it was observed that the total time was reduced. If the grid size is decreased, it will take more time with negligible difference in the stitching results so reaching the best suitable grid size for the data used to feed in the algorithm is critical.

Through successions of experiments, we reached the optimal grid size for our data which is 10 cm.

Finally, the root-mean-square error (RMSE) for the transformation defined as below in was considered as one of the evaluation criterions.

$$RMSE = \sqrt{\frac{\sum_{j=1}^{M} \sum_{i=1}^{N_j} \|\chi_j(a_i) - \hat{\chi}_j(a_i)\|^2}{\sum_{j=1}^{M} N_j}}$$

where $\chi_j(a_i)$ is the coordinate of a point a_i in the source cloud and $\hat{\chi}_j(a_i)$ the corresponding coordinate transformed from the reference cloud using the derived parameters *R* and *t*. The total number of points considered is the sum of the ones that can be paired in *M* clouds.

Using these varied combinations of data and extensive series of experiments, we reached the optimal values of the variants which gave the best stitching results.

4.1 Stitching Results

In the experiment, three busts of various features were used to populate sets of point clouds. The proposed method was used to reconstruct the datasets and perform stitching. The stitching results from the proposed augmented kNN and ICP method is shown in Figure 6, Figure 7 and Figure 8. It has three sub figures which depicts the solid models reconstructed from individual point clouds.

Figure 6

Stitching Result 1

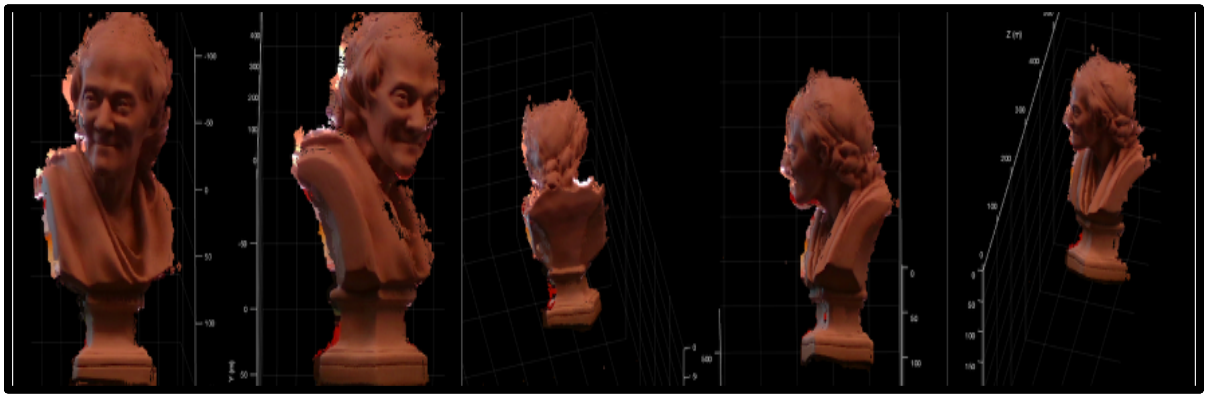


Figure 7

Stitching Result 2

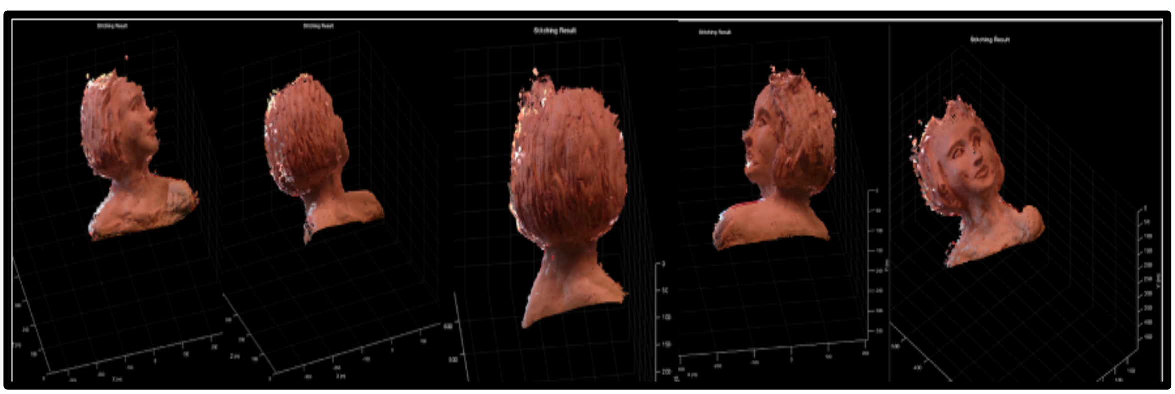
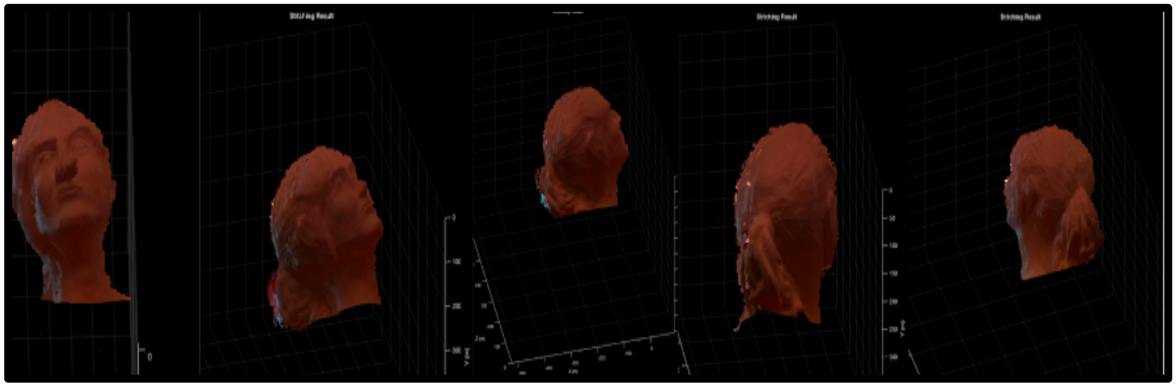


Figure 8

Stitching Result 3



4.2 Comparison with Standard ICP

In the experiment, three busts of various features were used to populate sets of point clouds.

First, the proposed kNNICP method was used to reconstruct them using the three datasets, Second, the same datasets are fed into the standard ICP For the comparison and to prove the efficiency of the proposed algorithm the thesis uses two criteria which are the average iterations per cloud and the root mean square error. The average iterations and RMSE for the transformation were then calculated for both standard ICP and Augmented ICP and KNN and comparison was made.

Table 1

Comparison with RMSE of Standard ICP and Modified ICP and KNN

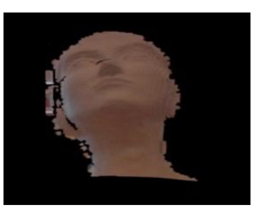
Objects	Attribute	Average Iteration per Cloud	Root Mean square Error
	ICP	18	3.6987
	ICP+kNN	16	1.9403
	ICP	20	3.9789
	ICP+kNN	14	2.4637
	ICP	25	4.9372
	ICP+kNN	20	2.4738

Table 1 shows the results of the comparison. As the table data shows that the average iterations per cloud is less when augmented ICP and KNN is performed which leads to less computation time whereas the standard ICP consumes more computation time. In comparison the observed values for root mean square error is also less for augmented ICP and KNN. Thus, it is clearly visible that the proposed algorithm has outperformed the basic ICP.

Figure 9

Comparison for RMSE

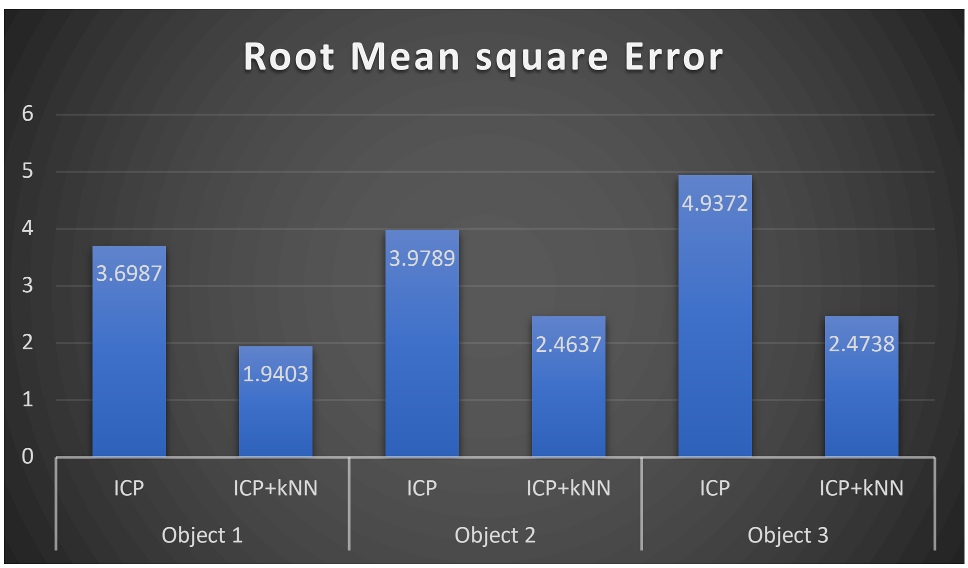


Figure 10

Comparison for Average Iteration per Cloud

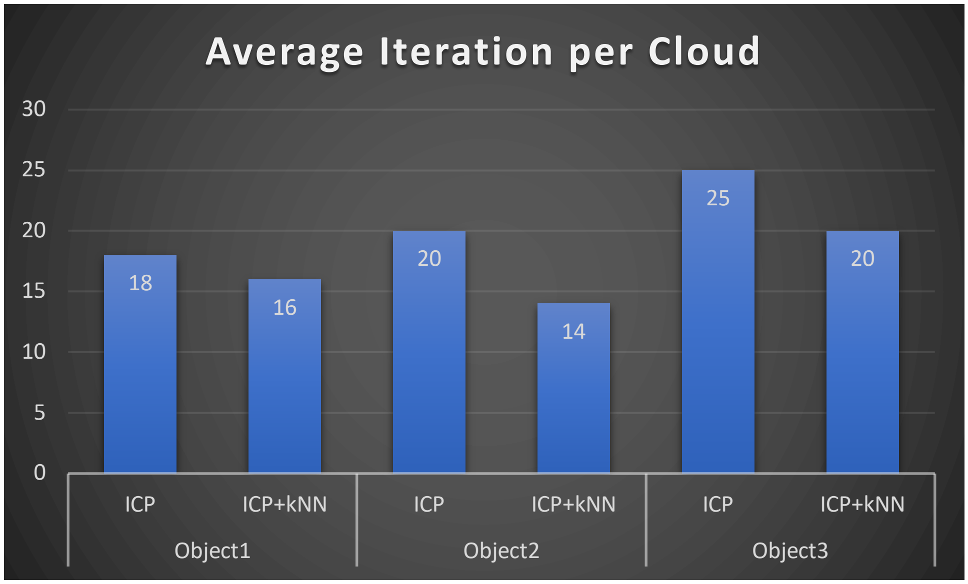


Figure 9 and Figure 10 depicts the graphical representation of the quantitative analysis performed for two categories that shows the performance ratios of the standard ICP and the proposed augment ICP with KNN. Augmented ICP with KNN has proved efficiency in both criteria's – root mean square error and average iteration per cloud.

Chapter 5

Conclusions

Recent trends in 3D modeling have highlighted the need for more accessible methods for stitching real objects into 3D models. To address the need for more accurate and efficient stitching methods, this paper proposes an incremental approach for 3D multi-angle point cloud stitching using an iterative closest point algorithm augmented with k-nearest-neighbors. With this combined algorithm, our method focuses on minimizing the error between neighboring point clouds, allowing us to easily compute the necessary transformation to combine point clouds into one model. Thus, when given concurrent point clouds captured at multiple angles of the same object, our approach provides a single accurate 3D model. We evaluated the ability of the proposed framework to stitch multiple point clouds into a solid model by stitching a segmented model and comparing the root mean squared error to a standard iterative closest-point stitching algorithm. From our experimental results, we first visually demonstrate the models obtained by stitching together a series of neighboring point clouds. Compared to other stitching methods, our proposed algorithm has reduced error and computational intensity because it does not need to reformat the data and instead operates directly on the 3D point cloud. Furthermore, our quantitative results demonstrate that kNN augmentation leads to a lower root mean-squared error compared to the standard iterative closest point algorithm. Overall, the proposed method is efficient, accurate, and robust for stitching 3D point clouds[1].

5.1 Recommendations for Future Work

In terms of quantitative analysis, additional parameters can be added for comparison between the ICP and the augmented Iterative closest point and K nearest neighbors to better prove the efficiency of the algorithm. Comparisons with the other stitching algorithms will also become part of Future work[1].

In terms of experimental analysis, future work will focus on implementing and testing algorithms with different types of objects with distinct surfaces like less textured and flat surfaces.

References

[1] P. Patel, R. Hare, Y. Tang and N. Patel, "3D Multi-Angle Point Cloud Stitching Using Iterative Closest-point Stitching and K-Nearest-Neighbors," 2022 International Conference on Cyber-Physical Social Intelligence (ICCSI), Nanjing, China, 2022, pp. 625-630, doi: 10.1109/ICCSI55536.2022.9970689.

[2] M. Kim, S. Kim and J. Choi, "Robust and incremental stitching and calibration with known rotation on pan-tilt-zoom camera," *2013 IEEE International Conference on Image Processing*, Melbourne, VIC, 2013, pp. 2247-2251, doi: 10.1109/ICIP.2013.6738463.

[3] P. Barrios and M. Adams, "Point set registration based on multi-object metrics," 2017 International Conference on Control, Automation, and Information Sciences (ICCAIS), Chiang Mai, 2017, pp. 245-250, doi: 10.1109/ICCAIS.2017.8217584.

[4] C. Yuan, X. Yu and Z. Luo, "3D point cloud matching based on principal component analysis and iterative closest point algorithm," 2016 International Conference on Audio, Language and Image Processing (ICALIP), Shanghai, China, 2016, pp. 404-408, doi: 10.1109/ICALIP.2016.7846655.

[5] Procházková, Jana & Martišek, Dalibor. (2018). Notes on Iterative Closest Point Algorithm.

[6] S. Taneja, C. Gupta, S. Aggarwal and V. Jindal, "MFZ-KNN — A modified fuzzy based K nearest neighbor algorithm," 2015 International Conference on Cognitive Computing and Information Processing (CCIP), 2015, pp. 1-5, doi: 10.1109/CCIP.2015.7100689.

[7] W. Hou, D. Li, C. Xu, H. Zhang and T. Li, "An Advanced k Nearest Neighbor Classification Algorithm Based on KD-tree," 2018 IEEE International Conference of Safety Produce Informatization (IICSPI), 2018, pp. 902-905, doi: 10.1109/IICSPI.2018.8690508.

[8] Y. Yu and H. Peng, "Automated high speed stitching of large 3D microscopic images," 2011 IEEE International Symposium on Biomedical Imaging: From Nano to Macro, 2011, pp. 238-241, doi: 10.1109/ISBI.2011.5872396.

[9] M. Hammond, A. Clark, A. Mahajan, S. Sharma and S. Rock, "Automated point cloud correspondence detection for underwater mapping using AUVs," OCEANS 2015 - MTS/IEEE Washington, 2015, pp. 1-7, doi: 10.23919/OCEANS.2015.7404431.

[10] X. Pan and G. Wang, "Parallax-tolerant image stitching based on mesh optimization," 2017 IEEE 2nd Advanced Information Technology, Electronic and Automation Control Conference (IAEAC), 2017, pp. 414-420, doi: 10.1109/IAEAC.2017.8054048.

[11] S. Wang, Y. Wang, M. Jin, X. D. Gu and D. Samaras, "Conformal Geometry and Its Applications on 3D Shape Matching, Recognition, and Stitching," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 29, no. 7, pp. 1209-1220, July 2007, doi: 10.1109/TPAMI.2007.1050.

[12] Bria, A., Iannello, G. TeraStitcher - A tool for fast automatic 3D-stitching of teravoxel-sized microscopy images. *BMC Bioinformatics* **13,** 316 (2012). https://doi.org/10.1186/1471-2105-13-316

[13] Y. Guo, H. Wang, Q. Hu, H. Liu, L. Liu and M. Bennamoun, "Deep Learning for 3D Point Clouds: A Survey," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 43, no. 12, pp. 4338-4364, 1 Dec. 2021, doi: 10.1109/TPAMI.2020.3005434.

[14] Y. Lin, Y. Huang, S. Zhou, M. Jiang, T. Wang and Y. Lei, "DA-Net: Density-Adaptive Downsampling Network for Point Cloud Classification via End-to-End Learning," 2021 4th International Conference on Pattern Recognition and Artificial Intelligence (PRAI), 2021, pp. 13-18, doi: 10.1109/PRAI53619.2021.9551070.

[15] Zhan Zongqian and Zhang Xiaoqian, "3D surface reconstruction and panorama stitching based on LCD-based calibration and multi-baselines stereo matching," 2009 International Conference on Image Analysis and Signal Processing, 2009, pp. 278-81, doi: 10.1109/IASP.2009.5054648.

[16] W. Huang, Y. Li, P. Wen and X. Wu, "Algorithm for 3D Point Cloud Denoising," 2009 Third International Conference on Genetic and Evolutionary Computing, 2009, pp. 574-577, doi: 10.1109/WGEC.2009.139.

[17] X. Shaowen, Y. Zhenyu and W. Weiyong, "Algorithm of denoising based on point cloud segmentation," 2010 5th International Conference on Computer Science & Education, 2010, pp. 1707-1711, doi: 10.1109/ICCSE.2010.5593560.

[18] B. Huang, Y. Tang, S. Ozdemir and H. Ling, "A Fast and Flexible Projector-Camera Calibration System," in IEEE Transactions on Automation Science and Engineering, vol. 18, no. 3, pp. 1049-1063, July 2021, doi: 10.1109/TASE.2020.2994223.

[19] B. Huang, Y. Tang, S. Ozdemir and H. Ling, "A Fast and Flexible Projector-Camera Calibration System," in IEEE Transactions on Automation Science and Engineering, vol. 18, no. 3, pp. 1049-1063, July 2021, doi: 10.1109/TASE.2020.2994223.

[20] The MathWorks Inc., “Ptcloudin,” Remove noise from 3-D point cloud - MATLAB. [Online]. Available: https://www.mathworks.com/help/vision/ref/pcdenoise.html. [Accessed: 20-Jun-2022].

[21] The MathWorks Inc., “Ptcloudin,” Downsample a 3-D point cloud - MATLAB. [Online]. Available: https://www.mathworks.com/help/vision/ref/pcdownsample.html. [Accessed: 20-Jun-2022].

[22] P. Kamencay, M. Sinko, R. Hudec, M. Benco and R. Radil, "Improved Feature Point Algorithm for 3D Point Cloud Registration," 2019 42nd International Conference on Telecommunications and Signal Processing (TSP), 2019, pp. 517-520, doi: 10.1109/TSP.2019.8769057.

[23] A. Makovetskii, S. Voronin, V. Kober and A. Voronin, "Regularized point-to-point and point-to-plane functionals in the point clouds registration problem," 2021 International Conference on Information Technology and Nanotechnology (ITNT), 2021, pp. 1-6, doi: 10.1109/ITNT52450.2021.9649208.

[24] D. Li, A. Wang, P. Ren and L. Wu, "An Allowance Optimal Distribution Method Based on Improved Iterative Closest Point Algorithm," 2018 10th International Conference on Measuring Technology and Mechatronics Automation (ICMTMA), 2018, pp. 515- 518, doi: 10.1109/ICMTMA.2018.00130.

[25] Q. Xie et al., "Precise Point Set Registration Algorithm for Indoor Scene Reconstruction," 2019 Chinese Automation Congress (CAC), 2019, pp. 4336-4339, doi: 10.1109/CAC48633.2019.8997195.

[26] Authors, "A Brief Survey on 3D Semantic Segmentation of Lidar Point Cloud with Deep Learning," 2021 3rd Novel Intelligent and Leading Emerging Sciences Conference (NILES), 2021, pp. 405-408, doi: 10.1109/NILES53778.2021.9600493.

[27] K. J. Weber and J. Seo, "A Mobile Robotic Application of Naive Multidirectional Stitching with SIFT," 2021 21st International Conference on Control, Automation and Systems (ICCAS), 2021, pp. 1889-1894, doi: 10.23919/ICCAS52745.2021.9649791.

[28] V. Renò, M. Nitti, M. di Summa, R. Maglietta and E. Stella, "Comparative analysis of multimodal feature-based 3D point cloud stitching techniques for aeronautic applications," 2020 IEEE 7th International Workshop on Metrology for AeroSpace (MetroAeroSpace), 2020, pp. 398-402, doi: 10.1109/MetroAeroSpace48742.2020.9160183.

[29] Nishino K, Ikeuchi K. Robust, “Simultaneous registration of multiple range images comprising a large number of points,” Electronics & Communications in Japan, vol. 87(8), pp.61-74, 2004.

[30] K. Dreczkowski and E. Johns, "Hybrid ICP," 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2021, pp. 6801-6808, doi: 10.1109/IROS51168.2021.9636600.

[31] B. Zhong and Y. Li, "Image Feature Point Matching Based on Improved SIFT Algorithm," 2019 IEEE 4th International Conference on Image, Vision and Computing (ICIVC), 2019, pp. 489-493, doi: 10.1109/ICIVC47709.2019.8981329.

[32] X. Chen, Q. Wu and S. Wang, "Research on 3D Reconstruction Based on Multiple Views," 2018 13th International Conference on Computer Science & Education (ICCSE), 2018, pp. 1-5, doi: 10.1109/ICCSE.2018.8468705.