

Rancang Bangun Rekonstruksi 3D Dengan *Kinect Xbox 360*

Trisni Wahyu Ningtias¹, Koko Joni¹, Riza Alfita¹

¹ Program Studi Teknik Elektro, Fakultas Teknik, Universitas Trunojoyo Madura
Jalan Raya Telang, Kabupaten Bangkalan, Jawa Timur 69162
E-mail: trisniwahyu03@gmail.com

ABSTRAK

Abstrak - Perkembangan teknologi yang semakin pesat dan efektif membawa dampak pada bidang teknologi. Salah satunya adalah pemindaian objek dengan menggunakan komputer. Pemindaian objek merupakan teknologi yang menggabungkan antara hardware untuk melihat objek dan software untuk mengolah data yang telah diterima oleh hardware. Proses manufaktur tradisional tanpa pemindaian 3D meliputi desain, analisis dan uji prototipe membutuhkan waktu yang sangat lama dan secara ekonomis mahal. Sedangkan dengan pemindaian 3D dinilai mampu lebih efisien dan praktis. Penelitian ini dilakukan untuk mempermudah rekonstruksi melalui proses pemindaian 3D dengan memanfaatkan sensor pada kamera Kinect 360. Objek akan secara langsung dipindai dengan kamera secara menyeluruh 360° yang dibagi menjadi 8 kali potret untuk mendapatkan data setiap sisinya. Hasil yang diperoleh akan diolah oleh sistem untuk selanjutnya diproses menjadi objek 3D. Proses pengambilan data objek menggunakan software Eclipse sedangkan pemutaran objek menggunakan motor stepper yang dikontrol arduino. Berdasarkan hasil pengujian dari penelitian yang telah dilakukan diketahui bahwa sensor infrared pada kamera Kinect kurang maksimal dalam memantulkan cahaya kembali pada benda yang memiliki permukaan tidak rata. Namun sebaliknya sensor bekerja dengan baik pada benda yang memiliki permukaan rata.

Kata kunci: objek, pemindaian, sensor, kamera, Kinect Xbox

Copyright © 2019 Universitas Muhammadiyah Jember.

1. PENDAHULUAN

Dewasa ini perkembangan teknologi yang kian canggih dan kompetitif membawa pengaruh pada bidang ilmu pengetahuan. Salah satu yang utama yakni pemindaian objek menggunakan perangkat komputer. Pemindaian objek merupakan perpaduan antara teknologi *hardware* untuk menangkap visual objek dan *software* sebagai sarana pengolah data yang diterima oleh *hardware* tersebut. Hasil visualisasi objek dua dimensi (2D) yang diterima dari kamera akan secara otomatis diolah oleh *software* menjadi objek tiga dimensi (3D).

Scanning 3D adalah proses menangkap informasi digital tentang bentuk objek dengan peralatan yang menggunakan kamera untuk mengukur jarak antara *scanner* dan objek. Hal ini dapat digunakan untuk kebutuhan modifikasi, manufaktur, monitoring yang dibantu komputer, atau hanya menyimpan informasi bentuk objek untuk kebutuhan di masa depan. Pada proses manufaktur tradisional terdiri dari beberapa tahapan yang meliputi desain, analisis dan uji prototipe. Proses ini dinilai memerlukan waktu yang cukup lama. Selain itu jika ditinjau dari segi ekonomi memerlukan biaya yang tidak sedikit. Apabila terdapat kesalahan pada tahap analisis hasil uji maka perlu dilakukan pengulangan proses awal yaitu desain. Hal tersebut tidak berlaku pada metode *scanning 3D* karena proses desain, analisis dan simulasi dapat dieksekusi secara *multitasking*, terintegrasi dan sangat cepat. Kemungkinan *error* yang terjadi dapat langsung dianalisa dan diperbaiki. Setelah perbaikan selesai dapat dilanjutkan ke tahap proses pengujian dan prototipe. *3D scanning* juga merupakan salah satu komponen dalam membangun sistem manufaktur CIM (*Computer Integrated Manufacturing* [1]).

Berdasarkan permasalahan tersebut, maka dilakukan sebuah penelitian mengenai pemodelan objek menggunakan *3D scanning* untuk memudahkan dalam memvisualkan objek secara lebih cepat untuk dunia manufaktur.

2. KAJIAN PUSTAKA

2.1 Objek 3D

3D atau 3 dimensi merupakan sebuah objek yang memiliki ukuran panjang, lebar dan tinggi berbentuk. Dalam matematika dan fisika 3d biasa disebut dengan bangun ruang. Namun ternyata 3D memiliki cakupan yang sangat luas termasuk pada bidang grafis, seni, animasi dan computer. Dimensi spasial menunjukkan bahwa terdapat suatu titik koordinat kartesius X, Y dan Z pada objek tiga dimensi. Sedangkan pemodelan merupakan pembentukan suatu benda atau objek. Berdasarkan objek dan basisnya tahap perancangan diproses menggunakan komputer. Melalui konsep dan proses desain, keseluruhan objek dapat diperlihatkan secara tiga dimensi, sehingga disebut sebagai pemodelan tiga dimensi [2].

2.2 Citra Digital

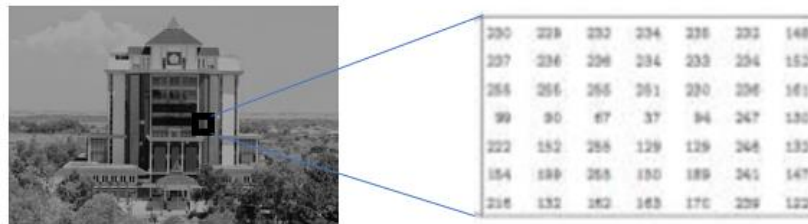
Citra merupakan intensitas fungsi 2 dimensi dari $f(x, y)$, dimana x dan y merupakan nilai dari koordinat spasial sedangkan f pada titik (x, y) merupakan tingkatan kecerahan citra pada titik tertentu. Citra digital merupakan hasil pengolahan citra yang dilakukan menggunakan alat bantu komputer. Jumlah bit yang digunakan untuk merepresentasikan setiap piksel pada masing-masing komputer mempengaruhi resolusi warna pada citra digital yang dihasilkan. Pada umumnya digunakan nilai 8-bit untuk merepresentasikan citra, 0 untuk hitam dan 255 untuk putih. Berikut merupakan jenis dari citra digital:

a. Black and White

Setiap piksel pada citra digital *black and white (grayscale)* memiliki warna yang berbeda mulai dari warna putih hingga hitam. Tingkatan nilai warna mengisyaratkan bahwa setiap piksel dapat diwakili oleh 8 bit, atau 1 byte. Rentang warna pada citra *black and white* sangat cocok digunakan untuk pengolahan file gambar. *Black and white* sebenarnya merupakan hasil rata-rata dari warna gambar, dengan demikian maka persamaannya dapat dituliskan sebagai berikut :

$$I_{BW}(x, y) = \frac{I_R(x,y) + I_G(x,y) + I_B(x,y)}{3} \quad (1)$$

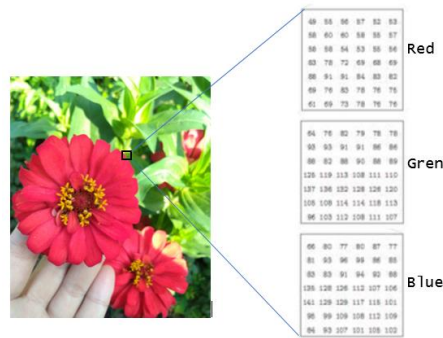
Pada persamaan diatas diketahui bahwa $I_R(x, y)$ = nilai *pixel red* titik (x, y) , $I_G(x, y)$ = nilai *pixel green* titik (x, y) , $I_B(x, y)$ = nilai *pixel blue* titik (x, y) , sedangkan $I_{BW}(x, y)$ = nilai *pixel black and white* titik (x, y) .



Gambar 1. Citra Black & White

b. Citra RGB

Citra RGB (*Red, Green, Blue*) adalah komponen warna pokok pada suatu gambar. Setiap titik piksel mempunyai warna tertentu yaitu merah (*Red*), hijau (*Green*), dan biru (*Blue*). Masing-masing warna mempunyai nilai ambang batas atau *range* 0 – 255, maka total keseluruhan dari seluruh warna adalah $255^3 = 16.581.375$ (16 K) variasi warna pada gambar. Pada setiap piksel memiliki tiga matriks yang berisi warna merah, hijau, biru [3].



Gambar 2 Citra RGB

2.3 Pengolahan Citra (*Image Processing*)

Pengolahan citra yang diproses menggunakan komputer awalnya dilakukan perubahan model citra menjadi beberapa bentuk besaran yang berbeda. Besaran tersebut terdiri dari bagian titik-titik citra yang memiliki tingkatan keabuan. Titik citra tersebut jika diinputkan pada layar monitor akan mengisi sebuah tempat yang biasa disebut piksel. Berdasarkan kemajuan teknologi komputer terdapat dua tujuan pengolahan citra yaitu:

1. Menyempurnakan nilai citra agar hasil citra dapat menampilkan data dengan jelas. Manusia mengambil peran sebagai pengolah data (*human perception*).
2. Mencari isi data yang sifatnya unik sebagai ciri pada citra secara numerik. Pada tahap ini komputer melakukan penafsiran pada data yang sudah ada agar dapat dibedakan menjadi data numerik secara lebih jelas [4].

2.4 Eclipse IDE

Eclipse merupakan suatu IDE (Integrated Development Environment) yang digunakan untuk proses pengembangan perangkat lunak agar dapat dijalankan di semua platform (platformindependent). Di bawah ini merupakan ciri-ciri dari Eclipse:

- a. Multi-platform : Sasaran pada sistem operasi Eclipse sangat beragam meliputi Microsoft Windows, Linux, Solaris, AIX, HP-UX dan Mac OS X.
- b. Multi-language : Eclipse umumnya dikembangkan menggunakan bahasa pemrograman Java namun tidak menutup kemungkinan *developer* menggunakan bahasa pemrograman lain seperti C/C++, Cobol, Python, Perl, PHP, dan lain sebagainya.
- c. Multi-role : Selain telah terintegrasi sebagai pengembangan aplikasi, Eclipse dapat digunakan untuk aktivitas dalam proses pengembangan perangkat lunak seperti dokumentasi, test perangkat lunak, pengembangan web, dan lain sebagainya.

Saat ini Eclipse menjadi salah satu IDE yang banyak digemari para pengembang karena sifatnya yang open source dan gratis. Setiap orang memiliki akses untuk dapat melihat kode pemrograman ini. Selain itu komponen plugin pada Eclipse menjadikannya semakin populer dikalangan penggunaannya [5].



Gambar 3. Logo Eclipse

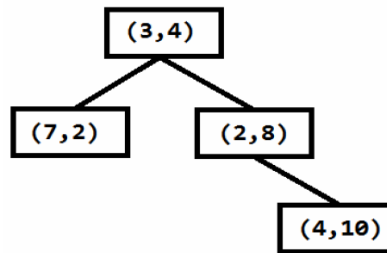
2.5 K-Nearest Neighbors

Metode *Nearest Neighbor* merupakan salah satu teknik klasifikasi yang berdasarkan kedekatan objek, dengan membandingkan jarak setiap objek. Pendekatan yang digunakan pada Nearest Neighbour sendiri merupakan pendekatan klasifikasi yang mencari semua data latih yang relatif mirip dengan data uji. Kedekatan disini didefinisikan dengan ukuran jarak, misalnya Euclidean. Jarak Euclidean antar dua titik, misal Titik1=(x1, y1) dan Titik2=(x2, y2) adalah :

$$Dist(Titik1, Titik2) = \sqrt{(x1 - x2)^2 + (y1 - y2)^2} \quad (2)$$

2.6 KD-Tree

KD-Tree (K-Dimensional Tree) merupakan representasi data multidimensional dalam bentuk binary tree yang bertujuan untuk memisahkan setiap data dalam suatu area tertentu berdasarkan nilai posisinya (Skiena, 2008). Contoh KD-Tree yang disusun dari data (3,4), (7,2), (2,8) dan (4,10) dapat dilihat pada Gambar 4 [6].

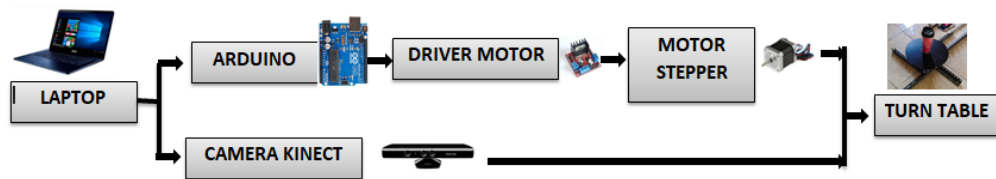


Gambar 4. Contoh KD-Tree

3. METODE PENELITIAN

3.1 Perancangan Perangkat Keras

Pada tahapan ini dirancang blok diagram cara kerja rekonstruksi 3D Kinect Xbox 360 berbasis mikrokontroler arduino. Alur proses pengolahan dijelaskan seperti pada gambar 5.

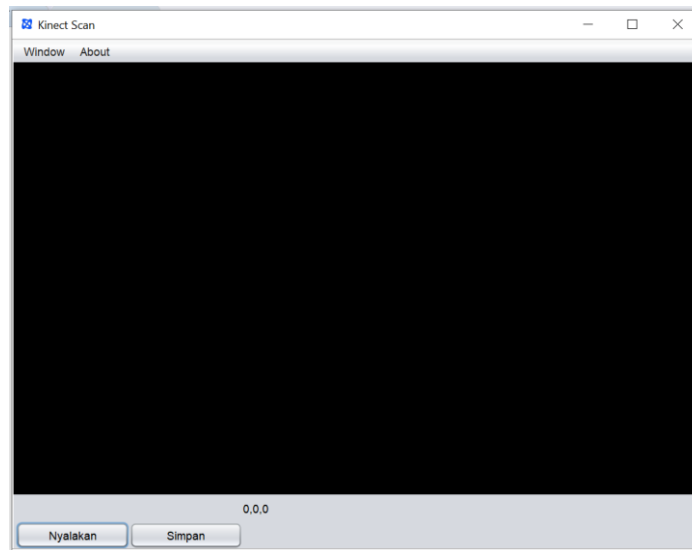


Gambar 5. Blok Diagram Hardware

Dari gambar 5 diketahui bahwa alur proses pengolahan dimulai dari PC yang tersambung ke arduino dan kamera kinect. Arduino sebagai mikrokontroler yang menerima output dari program arduino itu sendiri sedangkan kamera kinect digunakan untuk pengambilan objek gambar. Selanjutnya diantara arduino dan motor stepper dipasang driver sebagai switch berfungsi untuk melewatkan arus tegangan sesuai dengan kebutuhan motor stepper. Motor stepper di sini digunakan sebagai penggerak turntable sekaligus tatakan tempat benda yang akan dicapture dan direkonstruksi.

3.2 Perancangan Software

Gambar 6 merupakan hasil perancangan GUI Kinect Scan yang digunakan untuk mengambil data objek model. Kinect Scan disusun menggunakan software Eclipse Java yang di dalamnya terdapat dua tombol Nyalakan dan Simpan. Saat tombol Nyalakan ditekan maka tombol akan otomatis menjadi berubah kondisi menjadi aktif low begitu pula sebaliknya.

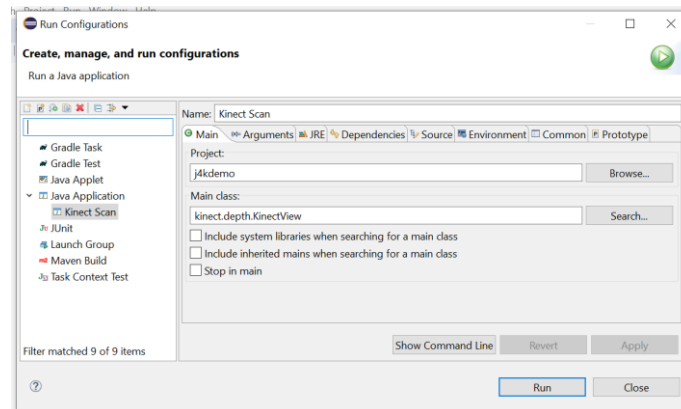


Gambar 6. Tampilan GUI Kinect Scan

3.3 Skenario Pengujian

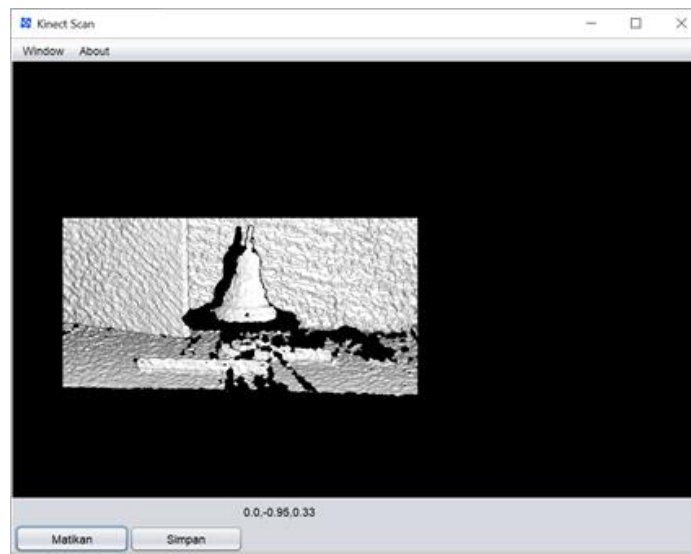
Pengujian kamera Kinect Xbox 360 dilakukan dengan cara menghubungkan usb kamera ke port usb pada laptop. Pengujian dilakukan dengan program Eclipse java. Prosedur pengujian yang dilakukan adalah sebagai berikut:

- a. Hubungkan usb kamera kinect ke port usb pada laptop dan adaptor kinect ke terminal
- b. Buka software Eclipse kemudian jalankan program KinectView dengan menggunakan configuration seperti pada gambar di bawah



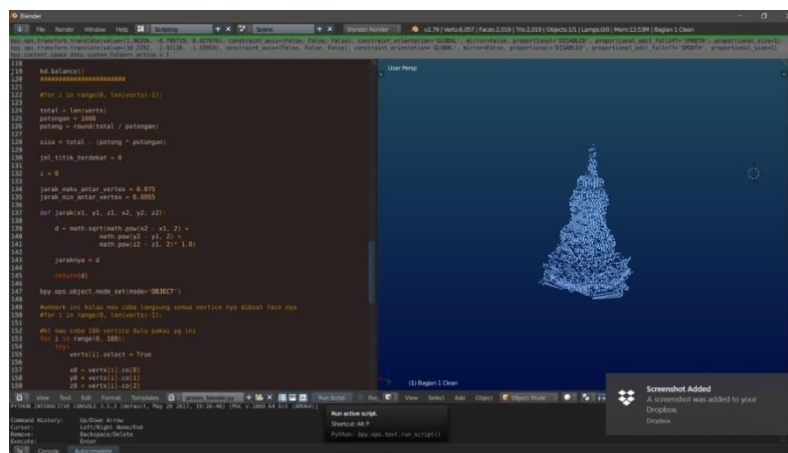
Gambar 7. Konfigurasi Program Kinect Scan

- c. Klik tombol nyalakan pada program yang sudah dijalankan
- d. Atur dan arahkan kamera ke objek yang akan diambil gambar
- e. Pastikan tampilan objek yang tertangkap kamera berwarna putih, jika masih berwarna hitam maka perlu dilakukan pengaturan jarak antara kamera dengan objek.



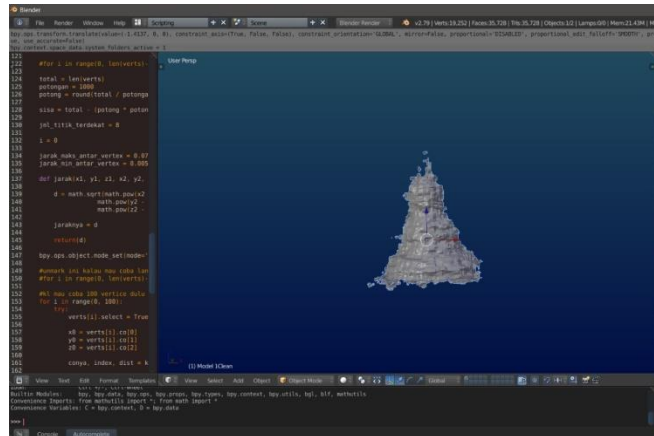
Gambar 8. Tampilan Objek Pada GUI

- f. Klik tombol simpan pada aplikasi, program akan menyimpan data depth yaitu depth_1.txt pada directory D: data_kinect
- g. Hasil data depth_1.txt yang telah disimpan selanjutnya diimport pada software Blender untuk diproses menjadi data mesh.
- h. Saat import file telah berhasil selanjutnya dilakukan proses meshing menggunakan script metode KD-Tree .



Gambar 9. Proses Meshing Data Depth

- i. Proses meshing menggunakan script metode KD-Tree menghasilkan data mesh.



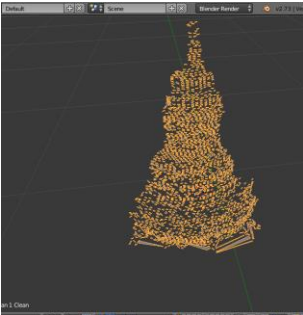
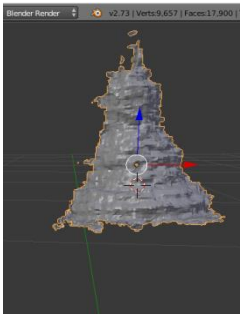

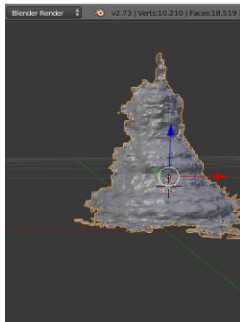
Gambar 10. Hasil Data Depth

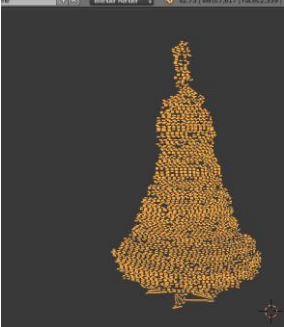
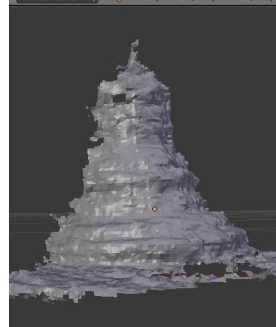
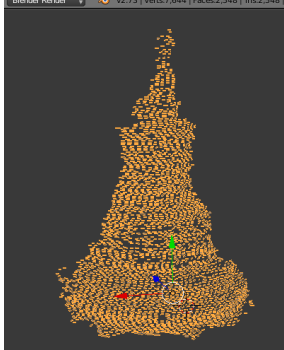
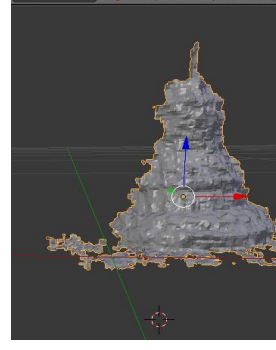
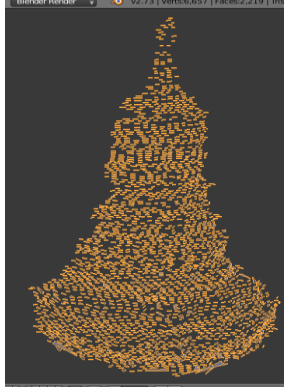
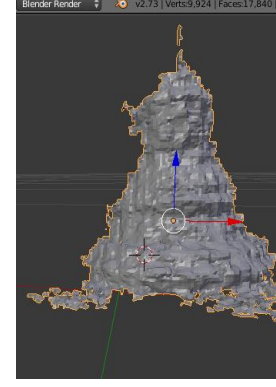

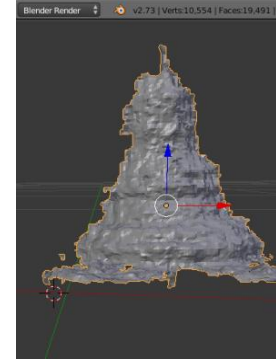
- j. Proses penggabungan masing-masing hasil mesh sehingga membentuk sebuah patung utuh.


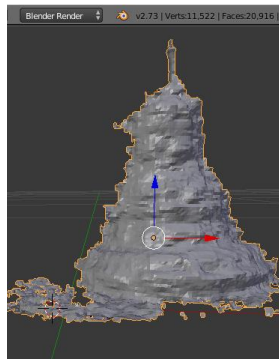
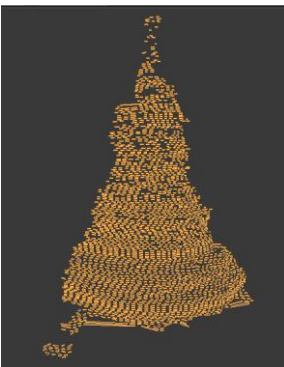
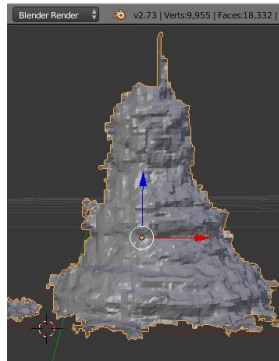
4. HASIL DAN PEMBAHASAN

4.1 Hasil Pengujian

Tabel 1. Hasil Capture Depth Data dan Hasil Mesh Model

Keterangan	Hasil Depth Data	Hasil Mesh
Capture ke-1 Sudut pandang 0°-45°		
Capture ke-2 Sudut pandang 46°-90°		

Keterangan	Hasil Depth Data	Hasil Mesh
Capture ke-3 Sudut pandang 91°-135°		
Capture ke-4 Sudut pandang 136°-180°		
Capture ke-5 Sudut pandang 181°-225°		
Capture ke-6 Sudut pandang 226°-270°		

Keterangan	Hasil Depth Data	Hasil Mesh
Capture ke-7 Sudut pandang 271°-315°		
Capture ke-8 Sudut pandang 316°-360°		

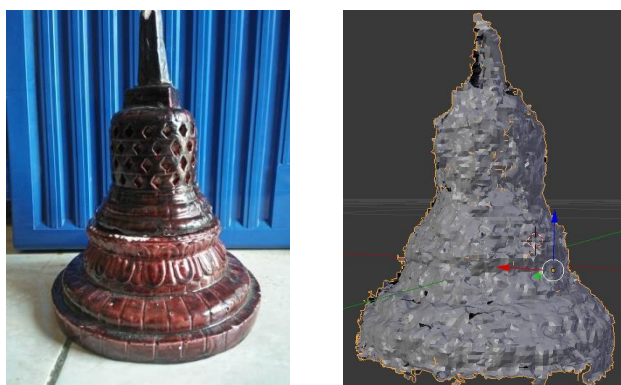
Pada tabel 1 Hasil Capture Depth Data dan Hasil Mesh Data terdapat dua kolom yang terdiri dari delapan kali percobaan berbeda. Masing-masing percobaan merupakan hasil dari sudut pandang setiap 45° mulai sudut 0° hingga 360°. Proses capture setiap bagian ditentukan oleh pergeseran sudut motor stepper. Motor stepper akan berputar sesuai dengan program yang telah dirancang pada software Arduino yang mana pada percobaan dirancang setiap satu kali putaran akan bergeser sebesar 45°.

Pada kolom hasil depth data merupakan representasi dari hasil capture camera kinect yang dilakukan menggunakan software Eclipse Java. Selanjutnya pada kolom hasil mesh data terdapat hasil olahan capture kamera kinect dengan menggunakan metode KD-Tree. Metode ini memproses model capture yang berupa puluhan ribu vertex menjadi sebuah model mesh yang solid. Setiap vertice yang berdekatan akan dikelompokkan oleh sistem pada software Blender berdasarkan nilai posisinya sehingga membentuk vertex. Kumpulan vertex-vertex yang telah tersusun akan membentuk sebuah model mesh data. Saat mesh data telah terbentuk maka perlu dilakukan proses cleaning up. Proses tersebut dilakukan untuk membersihkan bagian-bagian yang tidak diinginkan yang berada di sekeliling objek. Percobaan yang dilakukan pada model Patung Borobudur sekilas tampak seperti sama karena memang model ini memiliki permukaan sama. Akan tetapi setiap capturenya merupakan hasil dari delapan sudut pandang berbeda yang mana jika digabungkan menjadi satu akan membentuk sebuah patung utuh.

Berikut perbandingan model asli dengan hasil rekonstruksi 3D dari dua sudut pandang yang berbeda

1. Model Patung Candi Borobudur

a. Tampak Depan

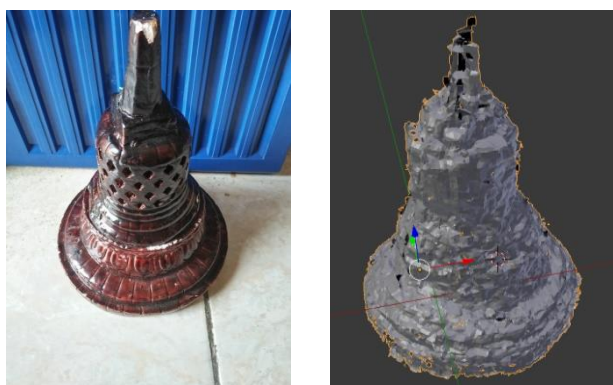


(a)

(b)

Gambar 11. (a) Model asli (b) Hasil Rekonstruksi

b. Tampak Atas



Gambar 12. (a) Model asli (b) Hasil Rekonstruksi

Pada gambar 11 Tampak Depan dan 12 Tampak Atas merupakan hasil percobaan yang menggunakan patung candi borobudur sebagai model objek. Patung tersebut dicapture dan direkonstruksi sesuai dengan langkah-langkah yang telah ditentukan. merupakan hasil depth data dan mesh dari objek model patung candi Borobudur. Model candi memiliki ukuran diameter dasar 17cm dan tinggi 23cm. Proses capture dilakukan sebanyak 8 (delapan) kali dengan pergerakan setiap sudut adalah 45° sehingga diperoleh nilai satu kali rotasi 360° . Posisi dari setiap sudut menghasilkan hasil depth image yang berbeda-beda sehingga hasil mesh yang diperoleh pun berbeda.

Proses capture sebanyak delapan kali berhasil dilakukan dan diperoleh data depth sesuai dengan step pengambilannya. Data depth yang diperoleh bergantung pada kondisi permukaan model yang digunakan. Data tersebut berisi koordinat xyz diproses menggunakan script python agar membentuk triangulasi. Rata-rata waktu yang diperlukan dari setiap model yang dicapture pada proses triangulasi adalah 50menit. Hasil triangulasi tiap capture akan diproses menjadi mesh menggunakan software blender dengan menggunakan script python sehingga membentuk model 3D. Pada objek ini diketahui terdapat lekukan dan lubang-lubang detail candi yang berukuran kecil. Permukaan yang tidak rata tersebut menyebabkan sinar infrared tidak terpantul secara maksimal. Hal tersebut berakibat pada hasil model 3D yang kurang detail sesuai model aslinya.

5. KESIMPULAN

Berdasarkan hasil empat percobaan yang telah dilakukan diperoleh kesimpulan sebagai berikut:

1. Permukaan objek yang akan direkonstruksi berpengaruh pada hasil akhir model 3D.
2. Pada permukaan yang tidak rata, infrared tidak bisa maksimal memantul kembali ke sensor. Sebaliknya pada objek dengan permukaan rata, infrared mampu memantul kembali ke sensor secara maksimal.
3. Semakin kecil ukuran objek yang dicapture maka semakin kecil pula tingkat kemiripan dengan objek aslinya.
4. Detail pada objek yang berukuran kecil dibawah 10cm tidak bisa direkonstruksi secara maksimal sehingga menyerupai objek aslinya.

DAFTAR PUSTAKA

- [1] E. Junaidi, Waslaluddin, and L. Hasanah, "Rancang Bangun Scanner 3D Menggunakan Sensor Ultrasonik dengan Tampilan Realtime Berbasis Mikrokontroler," *Fibusi J. Online Fis.*, vol. 3, no. 2, 2015.
- [2] P. Dimensi, U. Bangunan, and D. I. Kawasan, "The 3 Dimensional Modeling For Buildings At Puspiptek , Serpong," *J. Ilm. Geomatika*, vol. 18, no. 1, pp. 1–8, 2012.
- [3] R. Kastaman, Marsetyo, Sunarmani, and A. S. Somantri, "Aplikasi Pengolah Citra dengan Basis Fitur Warna 'RGB' untuk Klasifikasi Buah Manggis (Roni Kastaman dkk.)," *J. Bionatura*, vol. 10, no. 3, pp. 273–291, 2008.
- [4] Jogiyanto Hartono (1989:1), "Bab 2 landasan teori," *Apl. dan Anal. Lit. Fasilkom UI*, pp. 4–25, 2006.
- [5] M. Abdurrozzaq Almuzakki, "Rancang Bangun Aplikasi Location Based Service Pencarian Lokasi Wisata Di Kota Semarang Berbasis Android," *Dok. karya Ilm.*, pp. 1–8, 2013.
- [6] Y. Lukito, J. Wahidin, and S. Husodo, "Analisis Kinerja Struktur Data Kd-Tree Pada Metode K-Nearest Neighbors Kata kunci : KD-Tree , K-Nearest Neighbors , Array," *Riau J. Comput. Sci.*, vol. 2, no. 2, pp. 1–6, 2016.

BIOGRAFI PENULIS

	<p>Penulis bernama Trisni Wahyu Ningtias yang lahir di Madiun pada tanggal 08 Oktober tahun 1996. Saat ini sedang mengambil pendidikan S1 di prodi Teknik Elektro Universitas Trunojoyo Madura. Penelitian yang dilakukan oleh penulis yakni tentang pemodelan 3D dengan menggunakan Kinect. Penulis dapat dihubungi melalui email: trisniwahyu03@gmail.com</p>
	<p>Koko Joni, ST., M.Eng merupakan salah satu dosen tetap yang aktif mengajar di program studi Teknik Elektro Universitas Trunojoyo Madura. Lahir di Magetan pada tanggal 9 Juni tahun 1979. Penulis menamatkan pendidikan S1 Teknik Elektro di Insitut Teknik Sepuluh November pada tahun 2004. Pada tahun 2012 penulis telah berhasil menamatkan pendidikan S2 Teknik Elektro & Teknologi Informasi di Universitas Gajah Mada. Penulis dapat dihubungi melalui email: kokojoni@gmail.com</p>
	<p>Riza Alfita, S.T., M.T merupakan salah satu dosen tetap yang aktif mengajar di program studi Teknik Elektro Universitas Trunojoyo Madura. Penulis yang saat ini menjabat sebagai Koordinator Prodi Teknik Elektro lahir di Blora pada tanggal 19 April 1980. Penulis menamatkan pendidikan S1 Teknik Elektro pada tahun 2005 di Universitas Brawijaya. Pada tahun selanjutnya penulis mengambil pendidikan S2 Teknik Elektro di Universitas Gajah Mada dan menamatkannya pada tahun 2008. Penulis dapat dihubungi melalui email: riza.alfita@trunojoyo.ac.id</p>