

# Rancang Bangun Sistem Bantu Berjalan untuk Tuna Netra di Dalam Ruangan Berbasis Kamera

Muhammad Reza Imaduddin, Ronny Mardiyanto, Djoko Purwanto  
Teknik Elektro, Fakultas Teknologi Industri, Institut Teknologi Sepuluh Nopember (ITS)  
Jl. Kejawan Gebang, Surabaya 60111  
E-mail: ronny@elect-eng.its.ac.id

**Abstrak**— Penginderaan visual merupakan suatu proses pengolahan data visual menjadi data referensi yang dijadikan dasar untuk menggambarkan objek visual. Pada dasarnya penyandang tuna netra tidak mampu mengenali lingkungan sekitarnya, sehingga dengan bantuan tugas akhir ini maka penyandang tuna netra dapat mengetahui jika di sekitarnya terdapat halangan dan mampu berinteraksi dengan manusia lainnya. Dalam tugas akhir ini, dirancang sistem pengenalan pendeteksian halangan dan manusia untuk tuna netra di dalam ruangan berbasis kamera untuk mempermudah penyandang tuna netra mengenali lingkungannya khususnya untuk berjalan dengan aman tanpa tertabrak. Kamera diletakkan pada bagian atas helm pengguna dengan sudut pengambilan  $-20^\circ$  kebawah, ini bertujuan untuk mempermudah dalam proses pengambilan data visual. Kemudian data yang didapat diolah dengan beberapa metode pengolahan citra seperti *Depth Conversion*, *grayscale* dan *Human Detection* agar citra yang ditangkap oleh kamera dapat memandu ada atau tidaknya halangan dan manusia. Hasil pemrosesan data citra akan dikomunikasikan dalam bentuk suara melalui *earphone* pengguna. Pengujian dilakukan dengan pada waktu yang berbeda yakni pagi dan malam dan dilakukan dengan 3 cara yakni yang pertama pengujian pengukuran jarak, pendeteksian manusia, pengujian sistem keseluruhan. Sistem ini memiliki error rata-rata sebesar 20% dan akurasi rata-rata sebesar 80%. Error pengukuran disebabkan karena jarak tangkap efektif kamera yang digunakan pada sistem ini hingga 3 meter sehingga dengan jarak benda yang berada diatas 3 meter citra *depth* mengalami perubahan piksel yang menyebabkan benda tidak dapat terukur jaraknya.

**Kata kunci:** sistem bantu berjalan orang buta, kamera, Kinect, image processing, earphone

## I. PENDAHULUAN

Penyandang tuna netra memiliki banyak keterbatasan dalam menjalankan kehidupan sehari-hari. Keterbatasan itu didasari dari ketidakmampuan mereka dalam mengenali lingkungan secara baik. Misalnya mengenali benda-benda dan manusia di sekitarnya.

Untuk itu, dalam tugas akhir ini diimplementasikan rancang bangun sistem bantu berjalan untuk tuna netra, dimana alat ini akan diletakkan pada bagian atas tubuh pengguna. Kemudian dengan menggunakan teknik pengolahan citra (*Image Processing*) yang dilakukan oleh *Processing Unit* akan mempermudah pendeteksian halangan dan manusia terjadi, hasil pendeteksian halangan dan manusia akan dikomunikasikan dalam bentuk suara yang sesuai dengan kriteria-kriteria berjalan aman yang sudah diberikan. Sehingga penyandang tuna netra dapat secara aman dan cepat memenuhi kebutuhan dasarnya dan mengurangi ketergantungan orang lain.

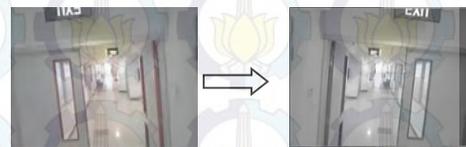
## II. TEORI PENUNJANG

### A. Pengolahan Citra (Image Processing)

Pengolahan citra adalah proses olah citra untuk mendapatkan informasi-informasi penting dari suatu objek yang nantinya akan diproses lebih lanjut sesuai dengan tujuan yang diinginkan. Pada bidang industri pengolahan citra dapat digunakan sebagai pengganti sensor, misalnya untuk menghitung jumlah botol dan memilah botol. Dengan menggunakan pengolahan citra dapat menggantikan beberapa sensor, sehingga lebih efisien, efektif dan hemat biaya.

Pada bidang militer pengolahan citra dapat digunakan untuk mendeteksi keberadaan musuh, sehingga dapat mengurangi kontak langsung dan korban jiwa. Salah satu metode pengolahan citra yang digunakan pada tugas akhir ini antara lain adalah *grayscale*. *Grayscale* adalah citra abu, bervariasi pada warna hitam pada bagian yang intensitas terlemah dan warna putih pada intensitas terkuat. Citra *grayscale* berbeda dengan citra "hitam-putih", dimana pada konteks komputer, citra hitam putih hanya terdiri atas 2 warna saja yaitu "hitam" dan "putih" saja. Pada citra *grayscale* warna bervariasi antara hitam dan putih, tetapi variasi warna diantaranya sangat banyak.

Citra *grayscale* disimpan dalam format 8 bit untuk setiap sample piksel, yang memungkinkan sebanyak 256 intensitas. Format ini sangat membantu dalam pemrograman karena manipulasi bit yang tidak terlalu banyak. Untuk mengubah citra berwarna yang mempunyai nilai matrik masing-masing R, G dan B menjadi citra *grayscale* dengan nilai X, maka konversi dapat dilakukan dengan mengambil rata-rata dari nilai R, G dan B. [1]

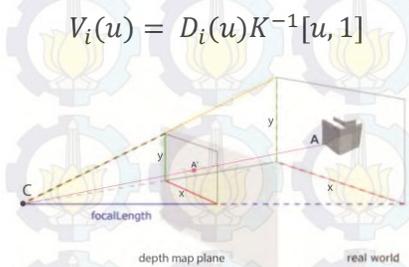


Gambar 1 Hasil grayscale

### B. Konversi Data *Depth*

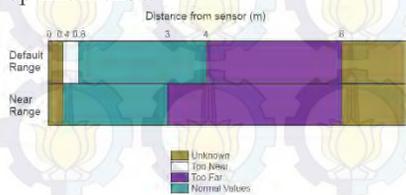
Dalam dunia grafika komputer dimensi tiga, peta kedalaman (*depth map*) adalah sebuah gambar dua dimensi yang berisi data kedalaman yang berisi informasi tentang jarak permukaan sebuah obyek terhadap sudut pandang (*view point*) tertentu. Jarak ini biasanya berhubungan dengan data kedalaman yang berupa koordinat Z dari sebuah koordinat kartesian. Dalam kasus ini, sumbu Z yang dimaksud adalah sumbu Z yang relatif terhadap view dari kamera, dan bukan merupakan sumbu Z dari koordinat dunia. Untuk melakukan

proses lebih lanjut, peta kedalaman harus dikonversi dari data dua dimensi menjadi data tiga dimensi yang memiliki sumbu X, Y, dan Z. Ilustrasi konversi peta kedalaman dapat dilihat pada gambar 2.5. Titik A diproyeksikan sebagai A' pada depth map plane. Dengan memperhitungkan focal length kamera, yaitu jarak antara pusat kamera (C) dan depth map plane, titik dalam dimensi dua pada depth map plane dapat dikonversikan menjadi titik dalam dimensi tiga. Konversi depth map ini melibatkan parameter intrinsik kamera yang disebut sebagai K dan juga melibatkan data depth map yang disebut sebagai D. Konversi ini dilakukan secara terpisah pada masing-masing titik di depth map yang dilambangkan sebagai u. Dengan kedua data tersebut, maka persamaan untuk melakukan konversi dapat dilihat pada persamaan 2-1. [2]

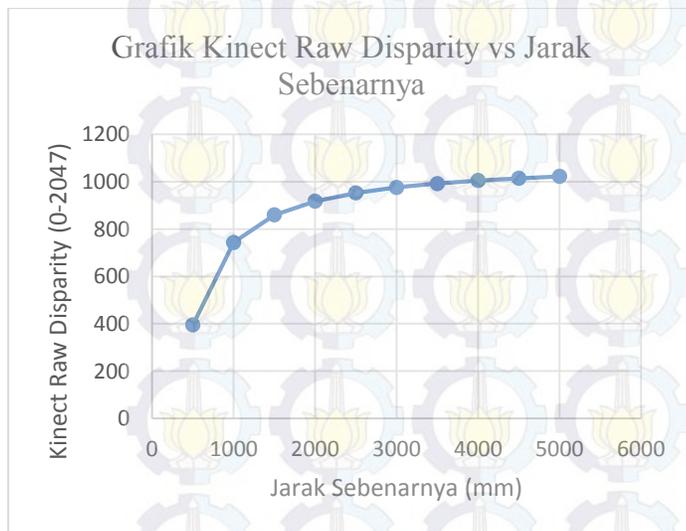


Gambar 2 Proses Konversi Data Depth

Lensa Depth ini memiliki ruang kerja yang optimal, sehingga nantinya dapat diketahui error yang akan terjadi berada pada jarak diluar optimal lensa.



Gambar 3 Depth Space Range



Gambar 4 Grafik Perbandingan Kinect Raw Disparity vs Jarak Sebenarnya

C. Human Detection

Dalam tugas akhir ini digunakan metode haar cascade, yang merupakan bagian dari metode viola jones. Metode viola jones adalah metode pendeteksi objek yang terdapat EmguCV, metode ini merupakan metode yang paling banyak digunakan untuk mendeteksi objek, hal ini dikarenakan metode viola jones memiliki algoritma yang efisien, sehingga tidak memerlukan waktu lama dalam melakukan proses pendeteksian objek. Teori ini menjumlahkan nilai-nilai dari beberapa citra (I) dalam sebuah kotak frame. Penjumlahan tersebut dapat dilihat dalam formula di bawah ini,

$$\sum_k \delta_k B_k(I)$$

dimana,  $\delta \in \{1, -1\}$  dan

$$B_k(I) = \sum_{i=u1(k)}^{u2(k)} \sum_{j=v1(k)}^{v2(k)} I_y$$

Beberapa fitur dapat dengan cepat di evaluasi oleh integral image, dimana formula integral image dapat di bawah ini,

$$I_y = \sum_{u=1}^i \sum_{v=1}^j I_{uv}$$

dimana u dan v adalah nilai dari pixel integral image. Maka dari itu penjumlahan nilai dari sebuah frame dapat dievaluasi dengan 4 tahap integral image. Maka dengan mudah kita dapat memeriksa bahwa

$$\sum_{i=u1}^{u2} \sum_{j=v1}^{v2} I_y = I_{u2v2} - I_{u1v2} - I_{u2v1} + I_{u1v1}$$

setiap fitur dari citra yang tertangkap dapat dievaluasi dengan metode integral image[6]. Algoritma deteksi upper body dapat dilihat pada tabel di bawah ini

Tabel 1 Algoritma deteksi upper body

1. Deteksi upper body
  - Membuat sebuah detektor objek cascade.
  - Baca frame video dan jalankan detektor.
  - Gambar sebuah kotak yang mengelilingi upper body yang terdeteksi.
2. Identifikasi fitur upper body untuk dilakukan pelacakan
  - Dapatkan informasi upper body dengan cara mengekstrak Hue dari frame video.
  - Ubah ke dalam bentuk warna Hue, Saturation and Value (HSV).
  - Tampilkan channel data Hue dan gambar kotak di sekitar upper body.
  - Deteksi upper body di sekitar area frame video.
3. Pelacakan upper body
  - Buat sebuah pelacak objek.
  - Inisialisasi pelacak histogram menggunakan channel pixel Hue.
  - Buat sebuah video untuk menampilkan frame video.
  - Lacak upper body di video hingga video selesai
  - Ekstrak frame video selanjutnya
  - Beri sebuah kotak mengelilingi upper body pada objek yang terdeteksi
  - Tampilkan dalam video yang telah diolah

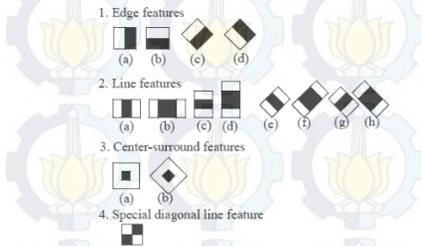
Proses pendeteksian objek dilakukan dengan mengklasifikasikan sebuah citra. Terdapat 4 hal tahap penting dalam teori viola jones, yaitu :

1. Fitur

2. Integral image
3. Adaptive boosting atau AdaBoost
4. Kombinasi Classifier of Cascade

❖ **Fitur**

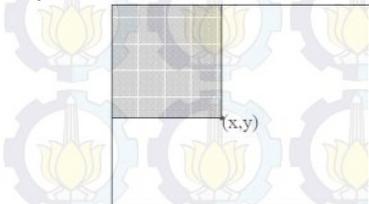
Fitur merupakan tahap paling awal yang diperlukan dalam pendeteksian objek. Penggunaan fitur dilakukan karena pemrosesan fitur berlangsung lebih cepat dibandingkan pemrosesan image per pixel. Fitur yang digunakan oleh viola-jones berdasarkan pada Wavelat Haar. Wavelat Haar merupakan gelombang tunggal bujur sangkar yang mempunyai satu interval tinggi dan satu interval rendah yang kemudian dikembangkan untuk pendeteksian objek visual yang lebih dikenal dengan fitur Haar atau fitur Haarlike.



Gambar 5 Fitur-fitur Haar (Lienhar, Kuranove, dan Pisarevsky)

❖ **Integral Image**

Integral image merupakan tahap kedua yang dilakukan dalam metode viola-jones. Integral image adalah sebuah image yang nilai tiap pixel-nya merupakan akumulasi dari nilai pixel atas dan kirinya. Sebagai contoh, pixel (a,b) memiliki nilai akumulatif untuk semua pixel (x,y) dimana  $x \leq a$  dan  $y \leq b$ .



Gambar 6 Nilai dari integral image pada titik (x,y) adalah jumlah dari semua pixel dari atas sampai kiri

Menurut viola dan jones integral image pada lokasi x,y berisikan jumlah pixel dari atas sampai kiri dari x,y, perhitungannya dapat dicari dengan menggunakan rumus di bawah ini :

$$ii(x, y) = \sum_{x' \leq x, y' \leq y} i(x', y')$$

dimana

$ii(x,y)$  = integral image

$i(x,y)$  = original image

lalu dengan menggunakan pasangan rumus berikut ini :

$$s(x, y) = s(x, y - 1) + i(x, y)$$

$$ii(x, y) = ii(x - 1, y) + s(x, y)$$

dimana  $s(x,y)$  adalah penjumlahan kumulatif baris,  $s(x-1) = 0$ , dan  $ii(-1,y) = 0$  sehingga integral image dapat dihitung dengan mengabaikan original image[7].

❖ **Adaptive Boosting**

AdaBoost merupakan tahap ketiga dalam metode viola-jones. Algoritma AdaBoost berfungsi untuk melakukan pemilihan fitur-fitur dalam jumlah banyak, dengan hanya memilih fitur-fitur tertentu. Boosting merupakan meta-algoritma dalam machine learning untuk melakukan supervised learning Untuk mendukung tujuan ini, algoritma pembelajaran yang lemah dirancang untuk memilih fitur persegi panjang tunggal, dimana persegi panjang tunggal merupakan yang terbaik untuk memisahkan contoh positif dan negatif. Untuk masing-masing fitur weak learner menentukan ambang batas klasifikasi fungsi yang optimal, sehingga jumlah minimum kesalahan sebuah pengklasifikasian yang lemah ( $h_j(x)$ ) terdiri dari fitur ( $f_j$ ), sebuah threshold ( $\theta_j$ ) dan kesamaan ( $p_j$ ) menunjukkan arah dari ketidaksetaraan tanda :

$$h_j(x) = \begin{cases} 1 & \text{jika } p_j f_j(x) < p_j \theta_j \\ 0 & \text{lainnya} \end{cases}$$

dimana x adalah sebuah 24 x 24 pixel sub-window dari sebuah gambar.

❖ **Kombinasi Classifier of Cascade**

Kombinasi Cascade of Classifier merupakan tahap terakhir dalam metode viola-jones. Dengan mengkombinasikan pengklasifikasian dalam sebuah struktur cascade atau cascade of classifier, kecepatan dari proses pendeteksian dapat meningkat, yaitu dengan cara memusatkan perhatian pada daerah-daerah dalam image yang berpotensi saja. Hal ini dilakukan untuk menentukan dimana letak objek yang dicari pada suatu image. Karakteristik dari algoritma viola-jones adalah adanya klasifikasi bertingkat. Klasifikasi pada algoritma ini terdiri dari 3 tingkatan dimana tiap tingkatan mengeluarkan subimage yang diyakini bukan objek. Hal ini dilakukan karena lebih mudah untuk menilai subimage tersebut bukan objek yang ingin dideteksi daripada menilai apakah subimage tersebut merupakan objek yang ingin dideteksi. Alur kerja klasifikasi bertingkat dapat dilihat di gambar 7 di bawah ini

D. **Perintah Suara**

Proses ini merupakan tahap akhir dalam sistem tugas akhir ini. Perintah suara pada sistem ini digunakan untuk mempermudah mengkomunikasikan hasil proses pengenalan simbol kepada pengguna. Keluaran suara dilakukan dengan menggunakan earphone satu sisi, sehingga suara yang dikeluarkan tidak mengganggu sekitar dan pengguna masih bisa mendengar lingkungan sekitarnya dengan telinga lainnya.

### III. PERANCANGAN SISTEM

Pada tugas akhir ini, sistem dilakukan di sebuah *processing unit* berupa *notebook*. Input data citra didapat dari sensor kamera berupa Kinect. Sensor kamera diletakkan pada bagian atas helm pengguna. Hal ini dilakukan untuk

mempermudah pengambilan data citra yang ada didepan pengguna. Kemudian dilakukan pengolahan citra gambar untuk mendeteksi ada tidaknya halangan dan manusia di depan pengguna. Ketika halangan dan manusia ditemukan maka akan dikomunikasikan dalam bentuk suara. Ilustrasi cara kerja sistem ditunjukkan pada gambar 8.

Berikut ini merupakan cara kerja keseluruhan sistem :

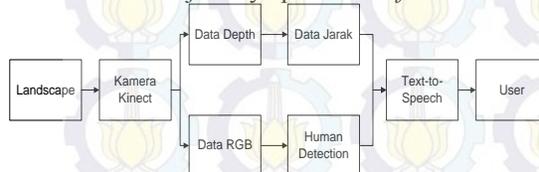
- Kinect menangkap data gambar dan kedalaman dari lorong
- Untuk data RGB yang ditangkap oleh Kinect dilakukan proses *smoothing* untuk menghilangkan noise dan detail-detail yang sangat kecil
- Kemudian diubah dari RGB direduksi menjadi citra grayscale untuk mempermudah proses pengolahan selanjutnya
- Kemudian dilakukan deteksi manusia yang memanfaatkan file tambahan dari *opencv*
- Untuk data Depth yang ditangkap oleh Kinect dilakukan proses konversi data Depth menjadi data millimeter
- Ketika halangan dan manusia terdeteksi di depan pengguna maka informasi akan disampaikan melalui suara



Gambar 7 Ilustrasi Cara Kerja Sistem

#### A. Perancangan Software

Tahapan ini terdiri dari beberapa tahapan yakni, pengambilan citra, *grayscale*, *surf*, membandingkan jumlah *keypoint*, perintah suara. Setiap tahapan memiliki fungsi masing-masing yang saling mendukung serta memiliki tahapan yang berurutan. Lebih jelasnya perhatikan *flowchart* berikut.



Gambar 8 Diagram Blok *Image processing* pada Sistem

#### B. Perancangan Hardware

Perancangan *hardware* pada sistem ini terdiri dari Kinect, *processing unit* dan *earphone*. Kinect dan *Earphone*

dihubungkan pada *notebook* dan dipasangkan kepada pengguna. Kamera menangkap citra, citra di proses oleh *notebook* dengan bantuan *image processing* setelah itu hasil yang didapat dikomunikasikan kepada pengguna melalui *earphone*. Berikut penjelasan tiap komponen yang dipakai dalam sistem tugas akhir ini.

#### 1. Kinect Sebagai Sensor Citra

Kinect yang digunakan adalah Kinect generasi pertama atau yang sering disebut dengan Kinect Xbox 360 dengan spesifikasi :

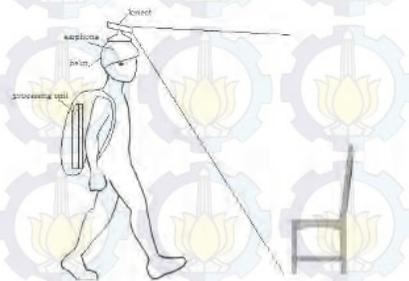
Kinect memiliki 1 buah lensa RGB dengan spesifikasi sebagai berikut :

- Resolution : 1600 x 1200 pixels
- Frame Rate : 30 frame per second

Kinect juga memiliki 1 buah lensa Depth dengan spesifikasi sebagai berikut :

- Resolution : 640 x 480 pixels
- Frame rate : 60 frame per second
- Operation Range : 0 - 3 m

Hal terpenting dalam pemilihan sensor citra pada sistem ini adalah kamera memiliki *image sensor* yang tinggi agar diperoleh gambar yang tajam, serta memiliki *frame rate* yang tinggi. Hal tersebut dilakukan untuk membantu proses pengolahan citra agar mengurangi beban *processing unit* serta kualitas citra yang baik akan menghasilkan akurasi yang baik pula. Kamera diletakkan di depan tubuh pengguna untuk mempermudah pengenalan simbol.



Gambar 9 Ilustrasi Pengambilan Data Citra

#### 2. Processing Unit

Pada sistem ini dipilih *notebook* sebagai *processing unit* karena beberapa alasan yaitu, *Notebook* bersifat *portable*, spesifikasi tinggi, memiliki ukuran yang cukup kecil, tidak terlalu berat. *Notebook* juga memiliki baterai bawaan yang cukup mampu digunakan berjam-jam dan memiliki banyak port *usb* yang penting untuk sistem ini.

Dimana spesifikasinya adalah sebagai berikut :

- Processor intel core i5 2.5 GHz Hyperthread (4 core)
- Ram 4 Gb
- Windows 7

Hal terpenting dalam pemilihan *processing unit* dalam sistem ini adalah kecepatan *clock processor* dan *memory ram*

yang mencukupi. Ini dikarenakan pengolahan citra membutuhkan memory ram yang besar dan pemrosesan yang cepat. Apabila di bawah standar, pemrosesan akan mengalami hambatan seperti *lagging* dan *break* saat proses berlangsung .

### 3. Earphone

*Earphone* digunakan untuk mempermudah mengkomunikasikan hasil pengenalan simbol kepada pengguna, pada sistem ini digunakan sennheiser mx 170 sebagai output suaranya.

Dimana spesifikasinya adalah sebagai berikut :

- Frekuensi 22-20.000 Hz
- Kepekaan 109 dB

## IV. PENGUJIAN

Pada tugas akhir ini, pengujian dilakukan untuk mengetahui kinerja dari sistem. Pengujian sistem ini dilakukan pada dua jenis lorong yang berbeda, yaitu lorong yang seluruhnya solid dan lorong yang pada salah satu sisinya berpagar (tidak solid) dan pada waktu yang berbeda, yaitu siang hari dan malam hari dengan pencahayaan dan tanpa pencahayaan sehingga tingkat kesalahan (error) dapat diketahui dan performa sistem yang optimal dapat diketahui.

### A. Pengujian Pengukuran Jarak di Dalam Lorong yang Seluruhnya Solid

Dengan menggunakan metode-metode yang sudah dijelaskan pada bagian disain dan implementasi, maka dilakukan pengujian terhadap lorong yang seluruhnya solid apakah sistem mampu menjaga user tetap berada dalam jarak aman untuk berjalan atau tidak. Dalam hal ini pengujian sudah dilakukan dengan dua jenis pencahayaan yang berbeda, yaitu menggunakan pencahayaan lampu dan tidak menggunakan cahaya lampu, berikut data-data yang didapatkan dari hasil pengujian

- Waktu : Siang Hari

Tabel 2 Tabel Pengukuran Jarak pada Jarak Optimal di Siang Hari

	Jarak Optimal Kinect dalam (m)					
Jarak	0.5	1	1.5	2	2.5	3
Hasil	0.511	0.978	1.458	1.952	2.434	2.93
Error(%)	2.2	2.2	2.8	2.4	2.64	2.33

Dari data di atas dapat disimpulkan bahwa jarak yang masih termasuk jangkauan dari Kinect dan jarak yang berada di luar jangkauan Kinect memiliki error yang sangat kecil sehingga tidak dapat membahayakan user dalam mendeteksi halangan pada siang hari dengan pencahayaan atau tanpa pencahayaan pada lorong yang seluruhnya solid.

- Waktu : Malam Hari

Tabel 3 Tabel Pengujian Jarak di Jarak Optimal Kinect dalam lorong yang seluruhnya solid pada Malam Hari

	Jarak Optimal Kinect dalam (m)					
Jarak	0.5	1	1.5	2	2.5	3
Hasil	0.511	1.022	1.496	1.986	2.487	2.98
Error(%)	2.2	2.2	0.267	0.7	0.52	0.67

Dari data di atas dapat disimpulkan bahwa jarak yang masih termasuk jangkauan dari Kinect dan jarak yang berada di luar jangkauan Kinect memiliki error yang sangat kecil sehingga tidak dapat membahayakan user dalam mendeteksi halangan pada malam hari dengan pencahayaan atau tanpa pencahayaan pada lorong yang seluruhnya solid.

### B. Pengujian Pendeteksi Manusia di Dalam Lorong yang Seluruhnya Solid

Dengan metode-metode yang sudah dijelaskan di atas pada bagian disain dan implementasi, maka dilakukan pengujian apakah sistem sudah mampu mendeteksi orang di dalam lorong yang seluruhnya solid pada siang hari untuk membantu *user* berinteraksi dengan lingkungan sekitarnya khususnya manusia. Pengujian dilakukan pada 2 buah pencahayaan yang berbeda yaitu, menggunakan pencahayaan lampu dan tidak menggunakan pencahayaan lampu, berikut adalah data-data yang didapatkan dari hasil pengujian

Tabel 4 Tabel Pengujian Deteksi Manusia di Siang Hari pada Jarak Optimal Kinect dengan pencahayaan lampu

	Jarak Optimal Kinect dalam (m)					
Jarak	0.5	1	1.5	2	2.5	3
Orang terdeteksi	-	√	√	√	√	√

Tabel 5 Tabel Pengujian Deteksi Manusia di Malam Hari pada Jarak Optimal Kinect dengan pencahayaan lampu

	Jarak Optimal Kinect dalam (m)					
Jarak	0.5	1	1.5	2	2.5	3
Orang terdeteksi	-	√	√	√	√	√

### C. Pengujian Sistem Keseluruhan Pada Subjek yang Berbeda

Setelah didapatkan data-data mengenai pengukuran jarak dan deteksi manusia, maka dilakukan pengujian sistem secara keseluruhan apakah sistem ini bisa dipakai secara *universal* sehingga kedepannya dapat berguna bagi para penyandang tuna netra. Berikut adalah hasil dari pengujian sistem secara keseluruhan :

- Subjek :

Tabel 6 Tabel Pengujian Sistem Secara Keseluruhan

No.	Peta	Keterangan
1.		<ul style="list-style-type: none"> <li>• Posisi awal dari <i>user</i> ketika akan melakukan pengujian sistem</li> </ul>
2.		<ul style="list-style-type: none"> <li>• Pada posisi ini, program mendeteksi ada halangan papan di depan dan memerintahkan <i>user</i> untuk berhenti dan mencari jalan lain</li> </ul>
3.		<ul style="list-style-type: none"> <li>• Pada posisi ini, <i>user</i> memilih jalan ke kiri dan berjalan hingga pada posisi terakhir pada peta ini program mendeteksi halangan lagi</li> </ul>
4.		<ul style="list-style-type: none"> <li>• Pada posisi ini, <i>user</i> berhasil melewati halangan manusia karena halangan tersebut masuk dalam kategori jarak terdeteksi halangan</li> </ul>

## V. KESIMPULAN

1. Kesalahan rata-rata dari pengukuran jarak pada waktu kapanpun menghasilkan nilai 2.2%, dari data tersebut dapat disimpulkan bahwa sistem ini tidak membahayakan *user* dalam melakukan aktifitas berjalan.
2. Performa metode ini sebenarnya tidak memiliki delay, satu-satunya delay yang membuat metode ini melambat adalah penyampaian melalui suara.
3. Hasil pengujian diperoleh pada pengujian sistem secara keseluruhan untuk beberapa user didapatkan error sebesar 20%, dikarenakan perbedaan interpretasi dari masing-masing user untuk memilih jalan mana yang akan dipilih.
4. Hasil pengujian diperoleh pada pengujian deteksi manusia didapatkan error sebesar 20%, hal ini dikarenakan

keterbatasan area jangkau dari frame Kinect dan besar Frame Per Second (FPS) yang dimiliki oleh Kinect.

5. Hasil pengujian diperoleh pada kecepatan berjalan terhadap ketepatan sistem didapatkan pengguna yang sudah pernah menggunakan sistem ini berjalan 1 langkah/detik sedangkan pengguna yang belum pernah menggunakan sistem ini berjalan 3 langkah/detik.

## DAFTAR PUSTAKA

- [1] Microsoft Developer Network Library. "Kinect for Windows Sensor Components and Specifications". <URL: <http://msdn.microsoft.com/en-us/library/jj131033.aspx>> Desember, 2010.
- [2] Izadi, Zahram. "KinectFusion: Real-time 3D Reconstruction and Interaction Using a Moving Depth Camera". Microsoft Research. Cambridge University, 2011.
- [3] S. Shao, "Mobility aids for the blind." in *Electronic Devices for Rehabilitation*. New York: Wiley, 1985, pp. 79-100.
- [4] Peter B. L. Meijer, "An Experimental System for Auditory Image Representations", *IEEE Trans. On Biomedical Eng.*, V01.39, N0.2 1992, pp.112-121
- [5] Bradski, Gary dan Kaebler, Adrian. "*Learning OpenCV Computer Vision with the OpenCV Library*", O' REILLY. 2008.
- [6] P. Viola and M. Jones, "Rapid Object Detection using a Boosted Cascade of Simple Features," in *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, 2001, pp. I-511-I-518 vol.1.
- [7] S. Ioffe and D. A. Forsyth, "Probabilistic Methods for Finding People," *International Journal of Computer Vision*, vol. 43, pp. 45-68, 2001.