

# Pengaruh Isolated Neighborhood-Averaging Filters Pada Kinect Structural Noise Sebagai Sistem Navigasi Robot Wild Thumper

Aditya Kurniawan, Kholilatul Wardani

Politeknik Kota Malang

Kompleks Pendidikan Internasional Tlogowaru No 3, telp/fax (0341) 754088

e-mail: aditya@poltekom.ac.id

**Abstrak** – Sensor kinect akan dikondisikan untuk mengambil sebuah benda hitam berbentuk persegi berukuran 10cm x 10cm secara tegak lurus pada sebuah background homogen (warna putih dengan RGB 255,255,255). Hasil data sensor kinect akan diambil menggunakan program visual basic 6.0 secara berkala sebanyak 10 kali tiap tahap dengan frekuensi satu kali / menit. Total tahap yang dilaksanakan adalah sebanyak 4 tahap. Sensor kinect diprogram untuk mengambil gambar objek tanpa proses blurring, dengan 1x blurring, 2x blurring dan 3x blurring menggunakan teknik Localized Neighbourhood-Averaging Filter. Hasil penginderaan sensor dari keempat sesi tersebut akan dianalisa berdasarkan tebal shadow (noise) pada tepi objek menggunakan statistik Pearson Correlation. Selanjutnya, mekanisme preprocessing sensor akan diujikan pada robot Wild Thumper.

Pengaruh Isolated Neighbourhood-Averaging Mask terhadap banyaknya pixel noise yang dihasilkan oleh kinect memiliki pengaruh dengan tingkat medium dengan koefisien Pearson bernilai 0,4, dan Layer titik jenuh Neighbourhood-Averaging filter pada pixel noise yang dihasilkan oleh kinect adalah dua layer.

**Kata kunci:** sensor kinect, wild thumper, structural noise, averaging filter

## 1. Pendahuluan

Saat ini, kebutuhan akan sistem elektromekanis otonom (biasa disebut robot) secara signifikan telah meningkat. Istilah otonom berarti mesin yang mampu melakukan tugas-tugas di dunia nyata tanpa kendali manusia secara eksplisit (Bekey, 2005: 1). Mesin otonom membutuhkan algoritma (metode berpikir) tertentu untuk mampu bertindak dengan cara tertentu pula. Selain algoritma, mekanisme penginderaan juga dibutuhkan untuk menjadikan mesin yang mati, menjadi sebuah sistem yang sadar akan lingkungan tempat dirinya beroperasi. Kedua sistem tersebut (mekanisme penginderaan dan algoritma) harus bekerja sedemikian rupa sehingga dapat bekerja secara independen tanpa kontrol dari manusia. Mesin otonom (robot) harus dilengkapi dengan beberapa sensor yang bekerja sama sebagai sebuah sistem penginderaan yang lengkap.

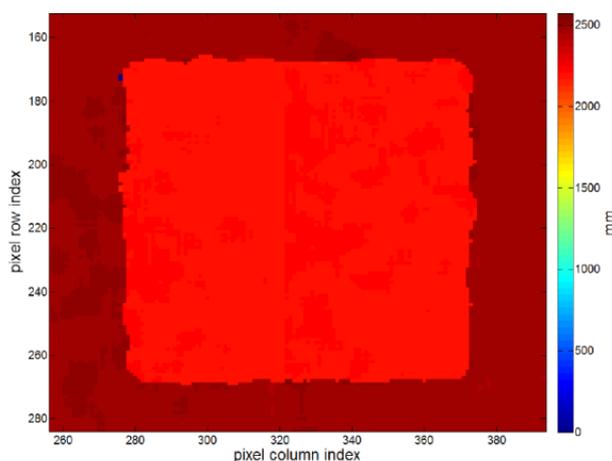
Beberapa penelitian (Durrant et al, 1987; Kuc et al, 1987; Han et al, 2009) telah mencoba untuk mengembangkan sensor yang ada, serta menciptakan algoritma baru untuk sistem sebuah robot; penelitian tersebut bervariasi dari sensor ultrasonik, sensor laser, sensor suhu, mikrofon array dan kamera. Salah satu contoh aplikasi sensor adalah untuk keperluan navigasi.

Sebuah studi kontemporer (Andersen, 2012) yang membahas mengenai vision sensor yaitu penelitian tentang kinect sebagai perangkat sensorik untuk robot. Kinect adalah sebuah vision sensor yang terdiri dari kamera cahaya tampak (visible light camera) dan sebuah sensor inframerah array. Sebuah kamera cahaya tampak dapat menangkap cahaya mulai dari 390 nm sampai 700 nm (dapat dilihat manusia). dengan mengukur berbagai parameter tergantung pada bagaimana algoritma diterapkan, sebuah kamera dapat memberikan robot sebuah penglihatan dua dimensi. Namun, kamera semacam ini tidak mampu mengukur jarak antara objek dan robot. Oleh

karena itu, kinect memiliki sensor tambahan berupa infrared array untuk mengukur jarak antara objek (hambatan) dan robot pada bidang spasial tertentu. Dengan demikian, sensor ini dapat memberikan robot informasi jarak (persepsi kedalaman) dan sekaligus penglihatan dua dimensi kepada robot.

Sensor infrared array bekerja dengan cara memancarkan inframerah kepada objek secara terus menerus (burst) dan secara acak. Sebuah kamera cahaya infra (infrared) kemudian menangkap radiasi inframerah yang menyentuh objek dan menentukan jaraknya berdasarkan pola radiasi pada objek. Emisi infrared yang ditembakkan secara acak ini membuat pembacaan pada infrared camera memiliki gangguan (noise) berupa bayangan pada tepi objek (Andersen et al., 2012)

Citra pada Gambar 1 menunjukkan tentang hasil pencitraan kinect pada sebuah objek berbentuk persegi. Tepi kotak secara fisik berupa garis lurus, tetapi karena terdapat structural noise pada sensor, hasil citra dari tepi objek tersebut terlihat tidak merata.



Gambar 1. Grafik Structural Noise Kinect Sensor

Noise semacam ini akan membuat robot melakukan kesalahan dalam menentukan bentuk objek yang menjadi sasaran. Variasi kesalahan ukuran objek karena noise diperkitakan antara 80 mm sampai 130mm tergantung oleh jarak antara sensor dan objek. Noise yang dihasilkan oleh Kinect dimungkinkan karena resolusi spasial sensor tidak cukup memadai untuk merekam pola IR dengan benar (Andersen, 2012).

## 2. Metodologi

### 2.1 Isolated Neighborhood Averaging Mask (INAM)

Istilah *Isolated pada filter Neighborhood-Averaging* berarti dalam pengaplikasiannya, sebuah *masking mean* (rerata) akan dijalankan pada titik / area kerja atau dinamakan dengan POI (*Point of Interest*). Dalam mengisolasi bagian yang akan di *filter* / di *masking*, digunakan *edge detection* untuk mensegmentasi daerah gambar yang akan diproses.

#### a. Segmentasi gambar

Proses ini dilakukan untuk mendapatkan tepi objek dari citra keseluruhan (tersegmentasi). Sebuah titik  $(i, j)$  dikatakan / didefinisikan sebagai tepi sebuah gambar jika titik tersebut memiliki nilai perbedaan yang tinggi dengan nilai *pixel* sekitarnya (Gonzalez, 2008).

Jika :

$P(i,j)$  : nilai *pixel* pada baris  $i$  dan kolom  $j$

$P'(i,j)$  : nilai *pixel* (hasil operasi) pada baris  $i$  dan kolom  $j$

$M(i,j)$  : filter yang digunakan untuk melakukan isolasi,

Maka:

$$P'(1,1) = \frac{1}{3} \times \sum_0^{i=2} \sum_0^{j=2} M(i,j) \times P(i,j) \quad (1)$$

b. *Masking* menggunakan teknik *Neighborhood-Averaging*

Proses ini dilakukan untuk mendapatkan nilai *edge* yang lebih halus (*smooth*). Sebuah titik / *pixel* yang memiliki perbedaan mencolok diasumsikan sebagai sebuah *noise*. Dengan melakukan proses *masking Neighborhood-Averaging(mean)*, diharapkan dapat menghilangkan *noise*.

Jika :

$P(i,j)$  : nilai *pixel* pada baris I dan kolom j

$P'(i,j)$  : nilai *pixel* (hasil operasi) pada baris I dan kolom j

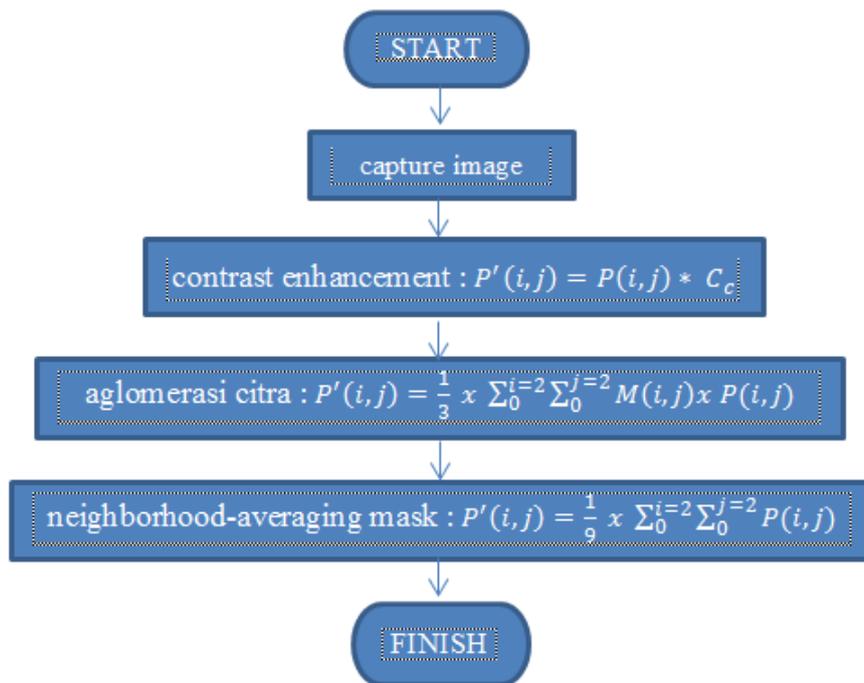
$M(i,j)$  : filter yang digunakan untuk melakukan *smoothing*,

Maka:

$$P'(1,1) = \frac{1}{9} \times \sum_0^{i=2} \sum_0^{j=2} P(i,j) \quad (2)$$

## 2.2 Algoritma Rancangan Penelitian

Perancangan dan pengaplikasian algoritma dilakukan dengan menggunakan pemrograman berbasis blok dan diagram (*labview*). Algoritma yang dirancang untuk diuji cobakan adalah sebagai berikut:

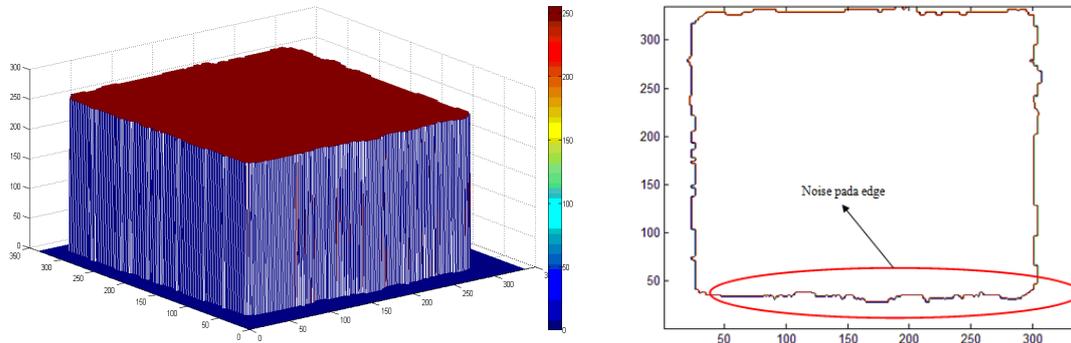


Gambar 2. Flowchart algoritma

Tahap pertama adalah melakukan *capture image* (pengambilan citra). Pada proses ini, membutuhkan sebuah aplikasi driver (Microsoft Kinect SDK) yang memungkinkan program *labview* untuk berhubungan dengan hardware.

### 3. Hasil dan Pembahasan

#### 3.1 Data capture Kinect sensor



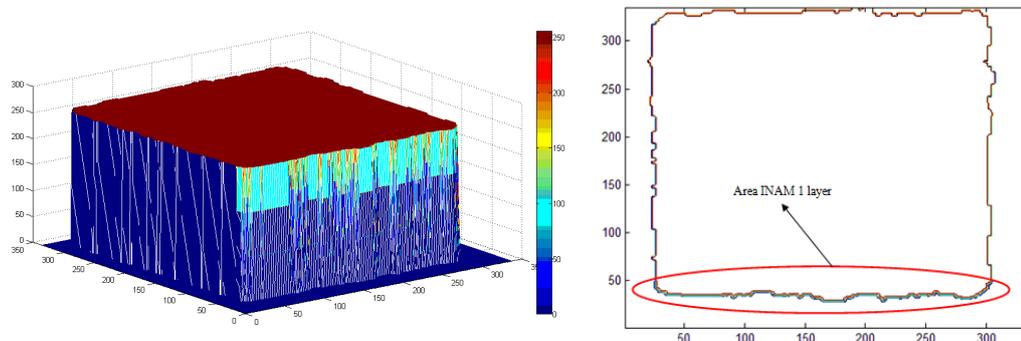
Gambar 3. Grafik data (3D dan 2D)

Data yang didapat dari kinect sensor menunjukkan bahwa garis tepi objek (edge) memiliki noise yang menyebabkan kesalahan data pengukuran bentuk atau ukuran objek seperti digambarkan pada grafik data elemen warna dalam entuk binary.

Pada Gambar 3 bagian kiri dapat dilihat bahwa representasi bentuk objek sebenarnya masih dapat terlihat, namun pada grafik contour yang berbentuk 2D, terlihat bahwa edge dari objek tidak rapi dan penuh dengan noise, seperti ditunjukkan pada gambar 3 bagian kanan.

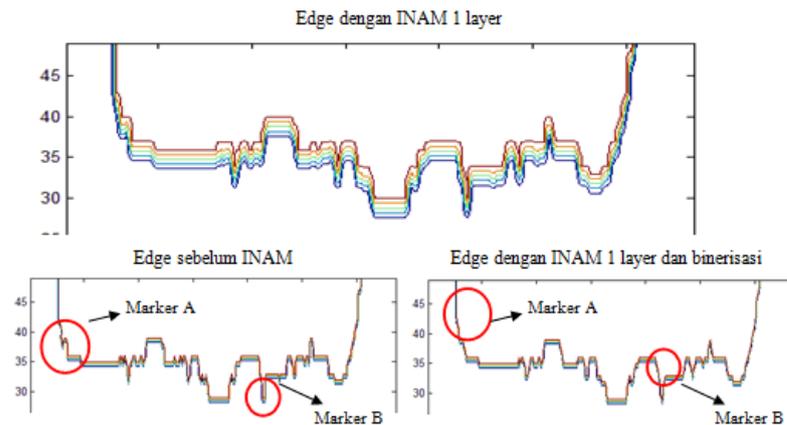
#### 3.2 Data capture Kinect sensor dengan averaging satu layer

Data yang didapat dari kinect sensor selanjutnya diproses dengan menggunakan INAM (*Isolated Neighborhood Averaging Mask*) dengan jumlah layer satu. Perlakuan filter hanya dijalankan pada edge yang dipilih sebagai POI (Point Of Interest) dari seluruh objek. Hasil dari filter tersebut dapat dilihat pada Gambar 4



Gambar 4. Grafik data (3D dan 2D) INAM satu layer

Dari data di atas dapat dibandingkan tebal edge sebelum dan sesudah INAM satu layer diberlakukan. Perbandingan tersebut disajikan pada Gambar 5.

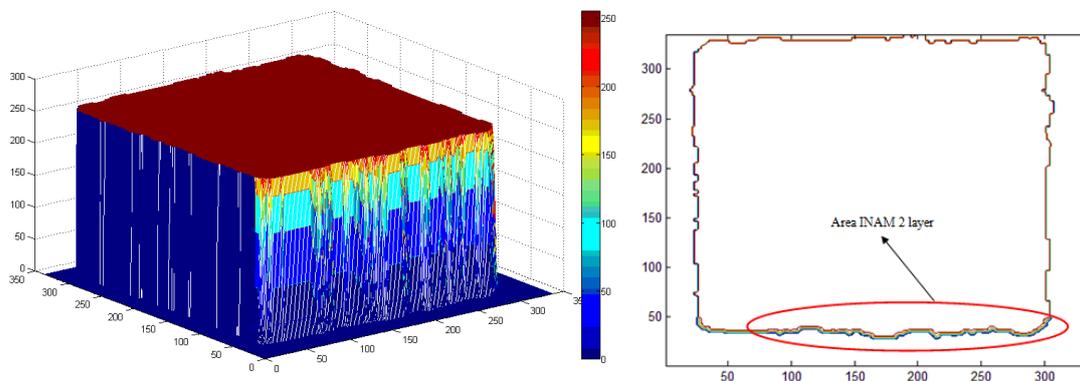


Gambar 5. Grafik edge INAM satu layer

Terdapat eliminasi noise dari pengamatan Gambar 5 pada marker A, dan pada marker B hanya sebatas pengurangan tebal (noise meruncing) memungkinkan untuk tereliminasi dengan konstanta contrast enhancement yang tepat.

### 3.3 Data capture Kinect sensor dengan averaging dua layer

Data yang didapat dari kinect sensor selanjutnya diproses dengan menggunakan INAM (*Isolated Neighborhood Averaging Mask*) dengan jumlah layer dua. Perlakuan filter seperti sebelumnya hanya dijalankan pada edge yang dipilih sebagai POI (Point Of Interest) dari seluruh objek. Hasil dari filter tersebut dapat dilihat pada Gambar 6.

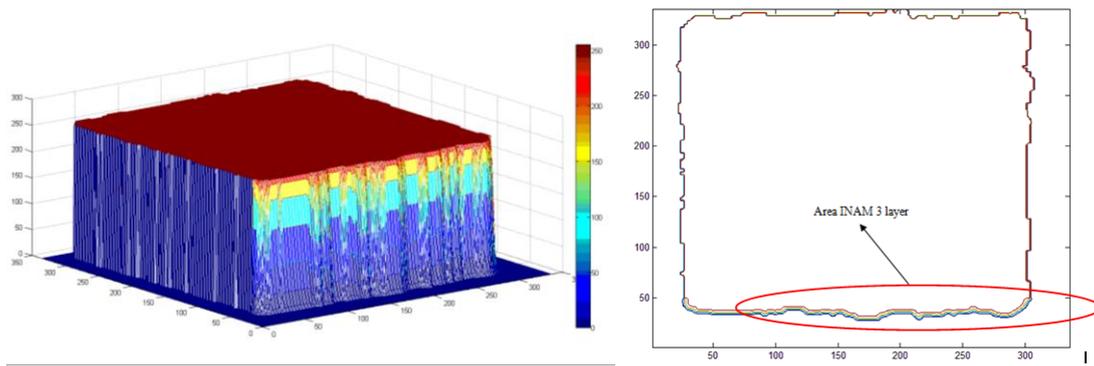


Gambar 6. Grafik data (3D dan 2D) INAM dua layer

Dari data di atas dapat dibandingkan tebal edge sebelum dan sesudah INAM dua layer diberlakukan. Perbandingan tersebut disajikan pada Gambar 6.

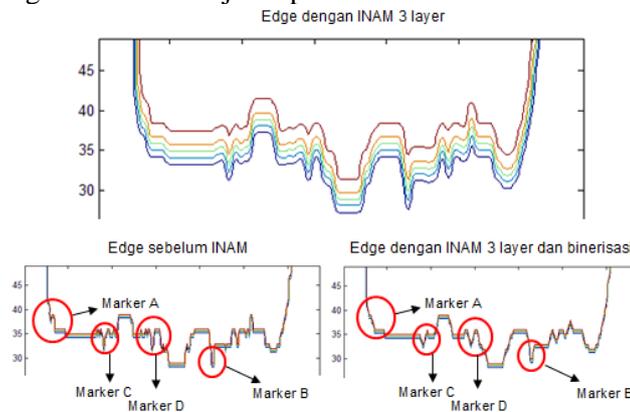
### 3.4 Data capture Kinect sensor dengan averaging 3 layer

Data yang didapat dari kinect sensor selanjutnya diproses dengan menggunakan INAM (*Isolated Neighborhood Averaging Mask*) dengan jumlah layer tiga. Perlakuan filter seperti sebelumnya hanya dijalankan pada edge yang dipilih sebagai POI (Point Of Interest) dari seluruh objek. Hasil dari filter tersebut dapat dilihat pada Gambar 7.



Gambar 7. grafik data (3D dan 2D) INAM tiga layer

Dari data di atas dapat dibandingkan tebal edge sebelum dan sesudah INAM 3 layer diberlakukan. Perbandingan tersebut disajikan pada Gambar 8.

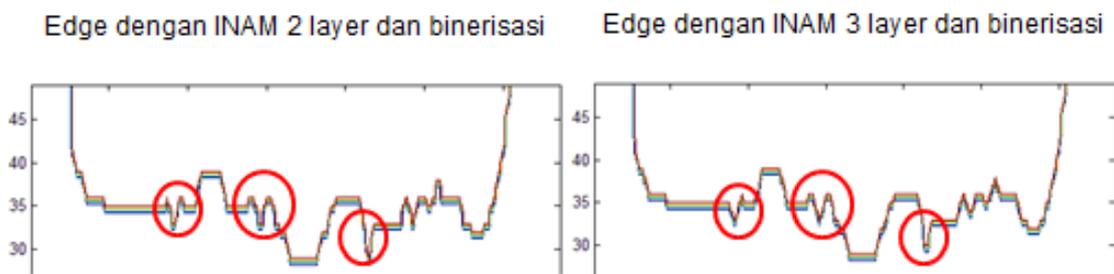


Gambar 8. Grafik edge INAM satu layer

Marker A, B, C, dan D tetap seperti pada layer dua. Tidak ada perubahan / penambahan area eliminasi.

### 3.5 Analisa titik jenuh Isolated Neighbourhood-Averaging Mask

Analisa titik jenuh layer dibutuhkan untuk mengetahui banyak layer yang dibutuhkan sampai INAM multilayer sudah tidak memiliki perbedaan signifikan dalam meng eliminasi noise. Perbandingan ditunjukkan pada Gambar 9.



Gambar 9. Grafik edge INAM satu layer

4. Penutup

4.1 Pengaruh Isolated Neighbourhood-Averaging Mask

Tabel 1 Parameter Pearson terhadap noise tiap layer

Banyak layer (x)	Tebal noise (y)	xi-x'	yi-y'	(xi-x')(yi-y')	(xi-x')^2	(yi-y')^2
0	13	-1,5	-0,5	0,75	2,25	0,25
1	14	-0,5	0,5	-0,25	0,25	0,25
2	13	0,5	-0,5	-0,25	0,25	0,25
3	14	1,5	0,5	0,75	2,25	0,25
Total				1	5	1

$$r_{xy} = \frac{n \sum (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{n \sum (x_i - \bar{x})^2} \sqrt{n \sum (y_i - \bar{y})^2}} \tag{3}$$

$$r_{xy} = \frac{1}{\sqrt{5} \sqrt{1}} = 0,4$$

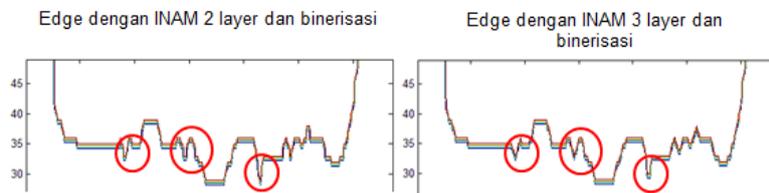
Tabel 2 Tabel Coeffisient Pearson

Strength of Association	Coefficient, r	
	Positive	Negative
Small	.1 to .3	-0.1 to -0.3
Medium	.3 to .5	-0.3 to -0.5
Large	.5 to 1.0	-0.5 to -1.0

Berdasarkan perhitungan pada persamaan  $r_{xy}$  dan Tabel 2, maka dapat disimpulkan Isolated Neighbourhood-Averaging filter terhadap banyaknya pixel noise yang dihasilkan oleh kinect memiliki pengaruh dengan tingkat *medium*.

4.2 Layer titik jenuh Neighbourhood-Averaging filter

Dalam hal ini, dua layer adalah level maksimum INAM, dengan layer tiga ke atas sebagai titik jenuh. Hal ini dapat ditunjukkan dari grafik INAM dua layer dan tiga layer setelah binerisasi yang tidak memiliki perbedaan signifikan seperti yang ditunjukkan pada Gambar 10.



Gambar 10. Grafik edge perbandingan INAM satu layer dan tiga layer

Perbedaan yang sangat kecil dapat dilihat pada tanda lingkaran merah. Dalam tahap ini, penambahan layer lebih dari titik jenuh tidak akan mengeliminasi noise pada area manapun. INAM dua layer dan tiga layer ke atas akan menampilkan grafik *contour* yang sama.

**Daftar Pustaka**

- [1] Andersen, M. R., Jensen, T., Lisouski, P., Mortensen, A. K., Hansen, M. K., Gregersen, T. and Ahrendt, P. 2012. Kinect depth sensor evaluation for computer vision applications.
- [2] Bekey, G. and Sukhatme, G. (2009). The path to autonomous robots. 1st ed. New York: Springer.
- [3] Castleman, K. (1979). Digital image processing. 1st ed. Englewood Cliffs, N.J.: Prentice-Hall.
- [4] Durrant-Whyte, H. (1987). Integration, coordination and control of multi-sensor robot systems. Kluwer Academic Publishers.
- [5] Gonzalez, R. and Woods, R. (2008). Digital image processing. 3rd ed. Pearson Prentice Hall.
- [6] Han, S., Park, S., Jung, J. and Lee, K. (2009). Mobile robot navigation by circular path planning algorithm using camera and ultrasonic sensor. pp.1749--1754.
- [7] Johnson, G. and Jennings, R. (2006). LabVIEW graphical programming. 1st ed. New York: McGraw-Hill.
- [8] Kramer, J. (2012). Hacking the Kinect. 1st ed. [New York]: Apress.
- [9] Kuc, R. and Siegel, M. (1987). Physically based simulation model for acoustic sensor robot navigation. Pattern Analysis and Machine Intelligence, IEEE Transactions on, (6), pp.766--778.