**MUHAMMAD RIZKY ANUGRAH**

**FISIKA B**

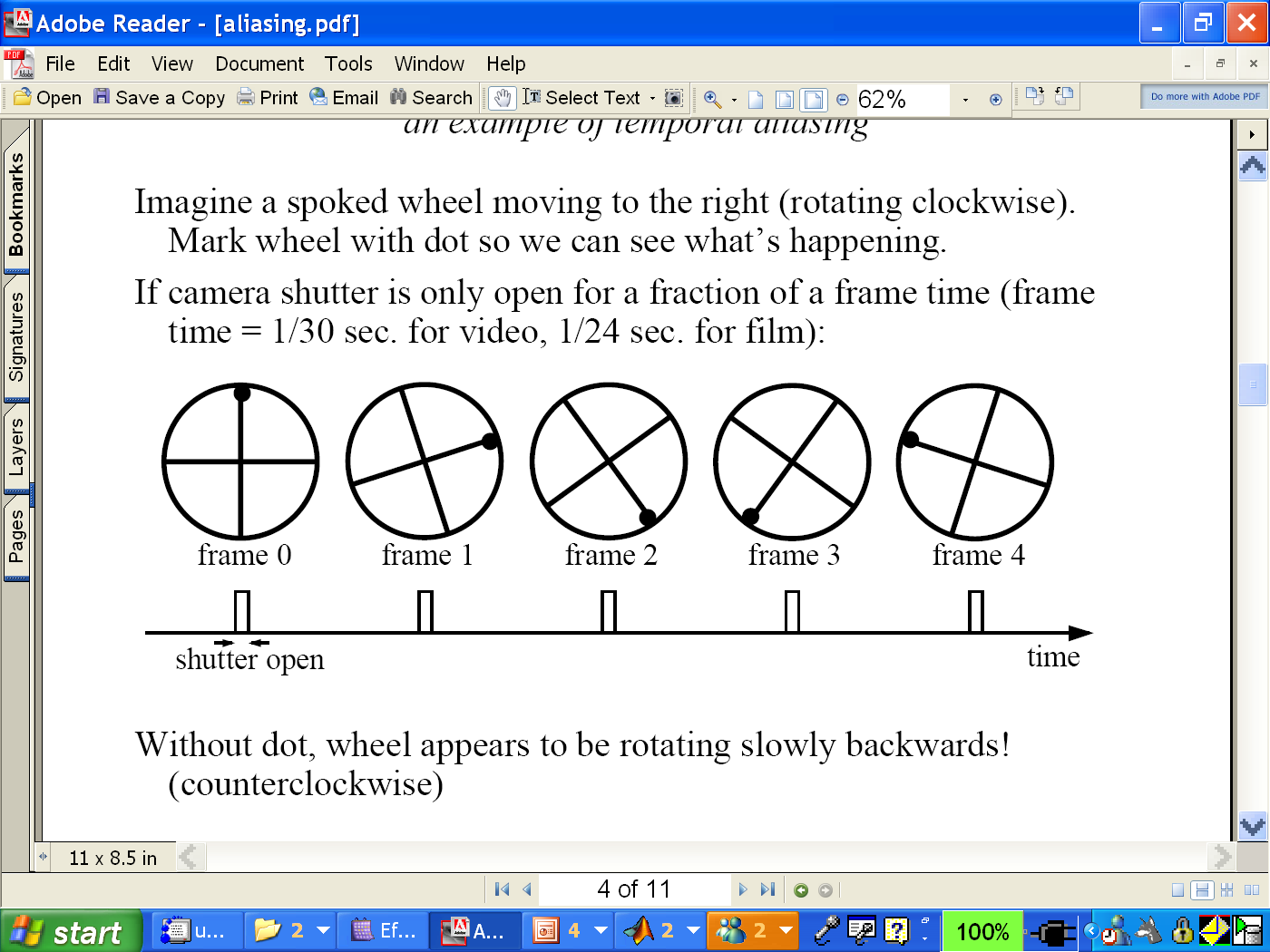
**1306620089**

**PENGOLAHAN CITRA DIGITAL**

**COMPUTER VISION**

**Aliasing - Sangat buruk dalam video**

Bayangkan roda spoked bergerak ke kanan (berputar searah jarum jam). Tandai roda dengan titik sehingga kita bisa melihat apa yang terjadi. Jika rana kamera hanya terbuka untuk sebagian kecil dari waktu bingkai (waktu bingkai = 1/30 detik untuk video, 1/24 detik untuk film):



tanpa titik, roda tampak berputar perlahan ke belakang! (berlawanan arah jarum jam)

**Deteksi Tepi**

* Mengkonversi gambar 2D menjadi satu set kurva.

1. Ekstrak fitur yang menonjol dari adegan.
2. Lebih kompak daripada piksel.

**Asal Tepi**

Tepi disebabkan oleh berbagai faktor

* Diskontinuitas normal permukaan
* Diskontinuitas kedalaman
* Diskontinuitas warna permukaan
* diskontinuitas iluminasi

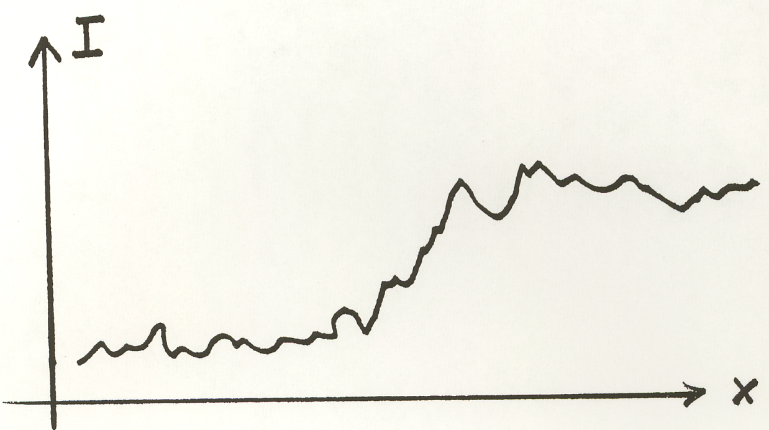
**Pada umumnya, hal yang ingin didapatkan dari operasi tepi yaitu:**

* Besaran Tepi
* Orientasi Tepi
* Tingkat Deteksi Tinggi dan Lokalisasi Bagus

**Kekuatan tepi diberikan oleh besaran gradien**

**Bagaimana kita bisa tahu bahwa piksel ada di tepi?**

1. **Jenis Tepi**
2. Tepi Langkah
3. Tepi Atap
4. Garis Tepi
5. **Tepi Nyata**

****

Kami menginginkan Operator Edge yang menghasilkan:

1. Besaran Tepi
2. Orientasi Tepi
3. Tingkat Deteksi Tinggi dan Lokalisasi Bagus

**Gradien**

1. Persamaan gradien:
2. Merupakan arah perubahan intensitas yang paling cepat.
3. Arah gradien:
4. Kekuatan tepi diberikan oleh besaran gradien

**Teori Deteksi Tepi**

1. Intensitas gambar (kecerahan)
2. Turunan parsial (gradien)
3. Gradien kuadrat
4. Laplasian

**Operator Tepi Diskrit**

Bagaimana cara membedakan citra diskrit?

1. Perkiraan perbedaan
2. Turunan parsial orde kedua
3. Laplasian

**Operator Sobel**

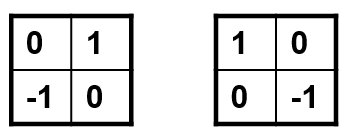
1. Perkiraan gradien yang lebih baik ada, Operator Sobel biasa digunakan.

Diagram

Description automatically generated

**Membandingkan Operator Tepi**

1. Gradien:
2. Roberts (2x2):

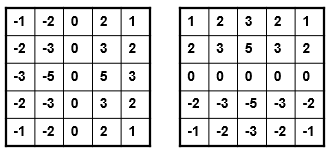
****

1. Sobel (3 x 3):

A picture containing crossword puzzle, shoji

Description automatically generated

1. Sobel (5 x 5):

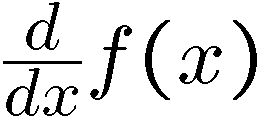
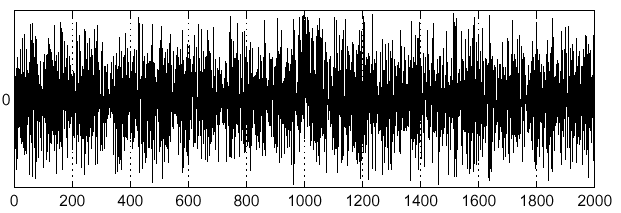
****

**Efek Kebisingan**

1. Pertimbangkan satu baris atau kolom gambar

Merencanakan intensitas sebagai fungsi posisi memberikan sinyal

****



Dimana Tepi??

Solusi

Calendar

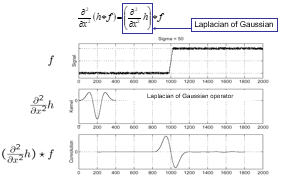
Description automatically generated

**Teorema Derivatif Konvolusi**

**Diagram

Description automatically generated**

**Laplasian dari Gaussian (LoG)**



**Operator Tepi Gaussian 2D**

1. Gaussian
2. Turunan dari Gaussian
3. Laplasian dari Gaussian

is the **Laplasian** operator:

**Operator Tepi Canny**

1. Gambar halus I dengan 2D Gaussian: *G\*I*
2. Temukan arah normal tepi lokal untuk setiap piksel 
3. Hitung besaran tepi 
4. Cari tepi dengan menemukan zero-crossings sepanjang tepi arah normal (penekanan non-maksimum) 

**Supresi Tidak Maksimum**

Periksa apakah piksel maksimum lokal di sepanjang arah gradientmembutuhkan pemeriksaan piksel yang diinterpolasi p dan r

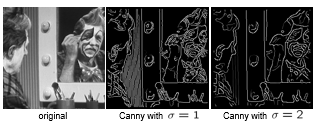
**Background pattern

Description automatically generated**

**Detektor Tepi Canny**

****

**Operator Tepi Canny**

****

* Pilihan tergantung pada perilaku yang diinginkan

1. Ukuran besar (1) mendeteksi tepi skala besar
2. Ukuran kecil (2) mendeteksi fitur-fitur bagus

**Deteksi Tepi DoG**

Deteksi tepi ini dilakukan dengan menggunakan difference of gaussian. Deteksi tepi ini didasari pada dua jenis gaussian yang berbeda yang dijadikan satu.



**Unsharp masking**

Metode ini dilakukan dengan tujuan mempertajam gambar yang blur. Metode ini membutuhkan 2 gambar yang sama dengan salah satunya memiliki tingkat blur yang lebih tinggi sehingga akan menghasilkan gambar yang lebih tajam. Adapun gambaran operasinya seperti gambar di bawah.

A collage of a person

Description automatically generated with low confidence

**Image Relaxation**

Metode ini merupakan deteksi tepi yang dilakukan dengan menggunakan iterasi dimana terjadi pengulangan beserta update nilai. Metode ini merupakan metode berulang untuk menyesuaikan tepi nilai berdasarkan tepi tetangga.

Adapun table actionnya yaitu:

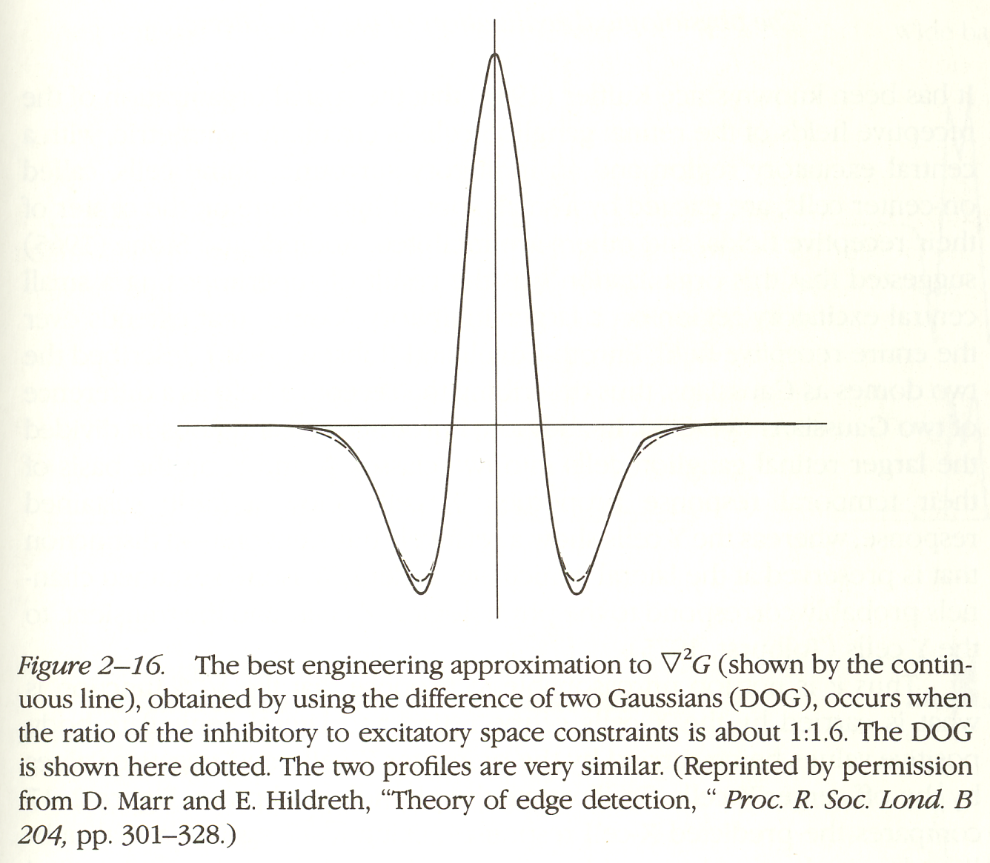
|  |  |  |  |
| --- | --- | --- | --- |
| Edge types | decrement | incremen | Leave as is |
| 0-0  0-2  0-3 | 1-1  1-2  1-3 | 0-1  2-2  2-3  3-3 |

Untuk algoritmanya yaitu:

1. Melakukan inisialisasi confidence untuk tiap edge *e* bernilai
2. Inisialisasikan k = 1
3. Lakukan konputasi edge type untuk tiap tepi *e*
4. Lakukan modifikasi confidence dengan basis dan edge type
5. Untuk melihat apakah sudah konvergen maka kembalilah ke step 3

**Perbedaan Gaussian**

Laplacian dari Gaussian dapat didekati dengan perbedaan antara dua Gauss yang berbeda.

****

**MATLAB**

g = fspecial('gaussian',15,2);

imagesc(g)

surfl(g)

gclown = conv2(clown,g,'same');

imagesc(conv2(clown,[-1 1],'same'));

imagesc(conv2(gclown,[-1 1],'same'));

dx = conv2(g,[-1 1],'same');

imagesc(conv2(clown,dx,'same');

lg = fspecial('log',15,2);

lclown = conv2(clown,lg,'same');

imagesc(lclown)

imagesc(clown + .2\*lclown)