

# Operasi Dasar pada Sinyal dan Citra

Pengolahan Sinyal Digital

Dr. Oddy Virgantara Putra

Department of Informatics  
Universitas Darussalam Gontor

24 May, 2025

# Overview

1. Operasi Penjumlahan
2. Operasi Penggeseran
3. Operasi Amplifikasi / Pembesaran
4. Superposisi
5. Sistem Linier
6. Sistesis
7. Dekomposisi

Kemampuan akhir yang diharapkan (Sub-CPMK):

1. Mahasiswa mampu mengidentifikasi dan menjelaskan penerapan teknologi sinyal dan citra digital dalam berbagai bidang.
2. Mahasiswa dapat melakukan operasi penjumlahan, penggeseran, dan amplifikasi pada sinyal diskrit

# Konsep Penjumlahan

Penjumlahan sinyal/citra adalah operasi pada tiap elemen.

Untuk sinyal 1D, apabila ada dua buah sinyal  $x_1[n]$  dan  $x_2[n]$ , kita bisa menjumlahkan dengan:

$$y[n] = x_1[n] + x_2[n] \quad (1)$$

Adapun untuk citra 2D, penjumlahan bisa dilakukan dengan:

$$I(i,j) = I_1(i,j) + I_2(i,j) \quad (2)$$

Penjumlahan ini bisa dilakukan dengan syarat:

1. Ukuran sinyal/citra harus sama
2. Tipe data juga harus sama

## Contoh Penjumlahan Sinyal 1D

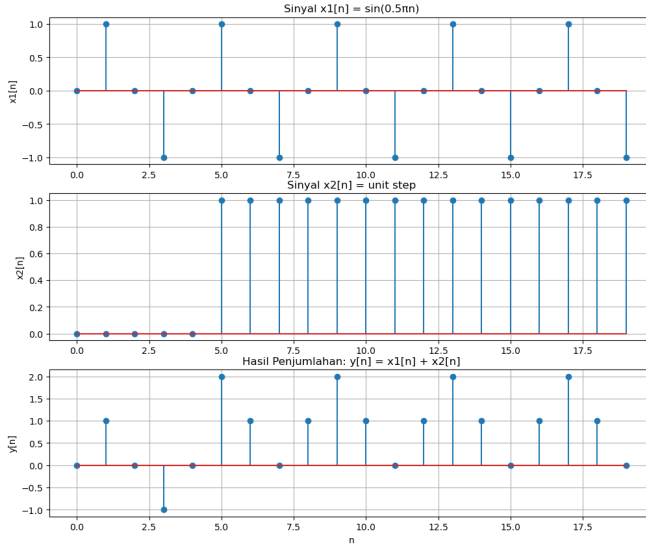
Contoh sinyal 1D:

$$x_1[n] = \sin(0.5\pi n) \quad (3)$$

$$x_2[n] = \begin{cases} 0 & n < 5 \\ 1 & n \geq 5 \end{cases} \quad (4)$$

$$y[n] = x_1[n] + x_2[n] \quad (5)$$

# Contoh hasil penjumlahan sinyal 1D



# Contoh Penjumlahan Citra

Contoh citra 2D:

$$I(i,j) = I_1(i,j) + I_2(i,j) \quad (6)$$

Image 1



Image 2



Added Image



# Pentingnya Penjumlahan dalam PSD

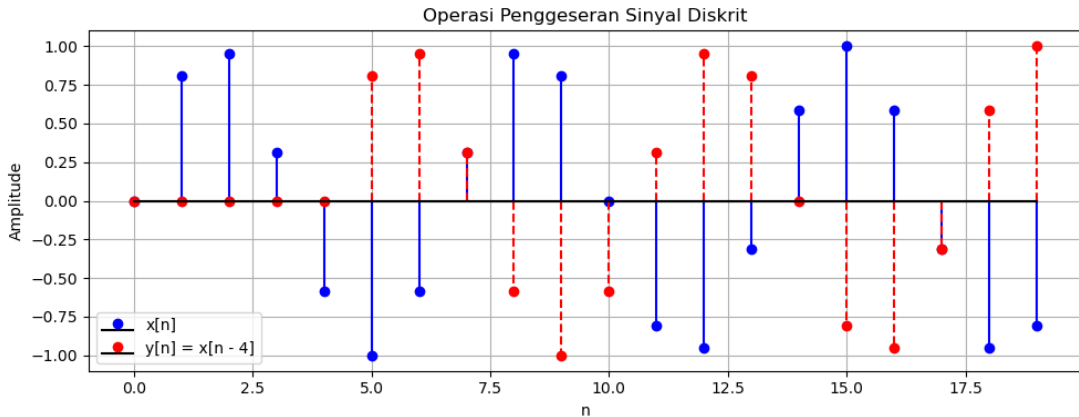
- Fondasi Superposisi
- Digunakan di banyak hal  
Audio mixing, Sensor fusion, Image blending, Efek visual
- Dasar operasi yang lebih kompleks  
Convolusi, Adaptive Filtering



# Penggeseran Sinyal Diskrit (1D)

Sinyal diskrit bisa digeser dengan:

$$y[n] = x[n - k] \quad (7)$$



# Penggeseran Citra 2D

Citra 2D bisa digeser dengan:

$$I'(i,j) = I(i - \Delta i, j - \Delta j) \quad (8)$$

Gambar Asli



Setelah Penggeseran



# Kegunaan Operasi Penggeseran

Dalam sinyal:

- Simulasi delay sistem (misalnya pada komunikasi)
- Time alignment

Dalam citra:

- Tracking objek yang berpindah
- Translasi dalam augmentasi data
- Pre-processing alignment antar frame pada video

# Amplifikasi Sinyal Diskrit (1D)

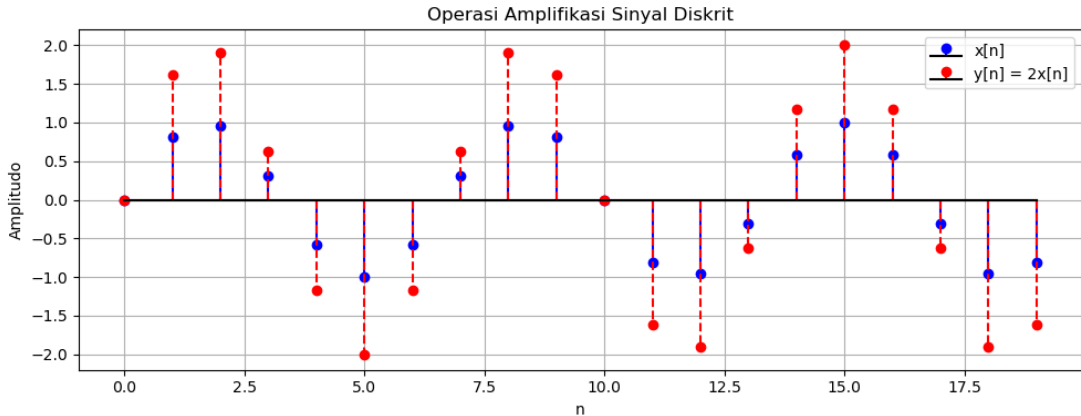
Sinyal bisa diamplifikasi/dikuatkan dengan persamaan berikut:

$$y[n] = \alpha x[n] \quad (9)$$

di mana  $\alpha$  adalah faktor pengali atau bisa disebut juga sebagai gain atau amplifier. Jika:

- $\alpha > 1$  maka akan memperbesar amplitudo
- $0 < \alpha < 1$  maka akan memperkecil amplitudo
- $\alpha < 0$  maka akan membalik fasa sinyal

# Contoh Amplifikasi sinyal 1D



Citra bisa diamplifikasi/dikuatkan dengan persamaan berikut:

$$I'(i,j) = \alpha I(i,j) \quad (10)$$

di mana  $\alpha$  adalah faktor pengali atau bisa disebut juga sebagai gain atau amplifier. Jika:

- $\alpha > 1$  maka akan meningkatkan brightness/kontras
- $0 < \alpha < 1$  maka akan mengurangi kecerahan

# Contoh Amplifikasi Citra 2D

Citra Asli



Citra Setelah Amplifikasi ( $a=1.5$ )



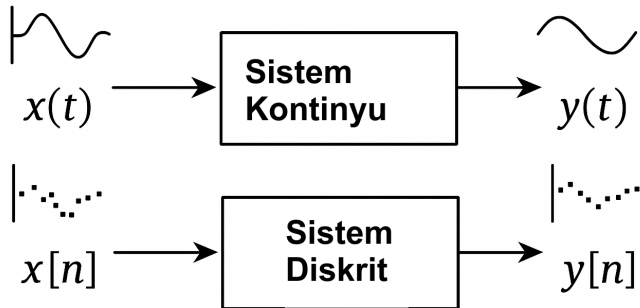
Menariknya, operasi-operasi seperti penjumlahan dan amplifikasi bukan sekadar manipulasi numerik—mereka mencerminkan sifat sistem yang lebih fundamental: superposisi [Oppenheim, 1967].



Sebagian besar teknik Pengolahan Sinyal Digital (PSD), didasarkan pada strategi superposisi. Pada prinsipnya, sinyal yang akan diproses, dipecah menjadi komponen-komponen yang lebih sederhana. Pemrosesan komponen secara individu ini nanti akan digabungkan kembali. Dari sinyal yang rumit, dipecah menjadi sinyal yang lebih banyak tetapi lebih mudah [Smith, 1997].

# Sistem Linier

Sistem adalah segala proses yang menghasilkan output sebagai bentuk respon atas input. Sistem Linier adalah sebuah sistem yang memiliki dua properti utama, homogenitas dan skala [Sayood, 2012]. Adapun Smith, menyebut komponen kedua sebagai unsur *additivity* [Smith, 1997]



# Homogenitas

Homogenitas pada PSD didefinisikan sebagai perubahan amplitudo pada sinyal input menghasilkan perubahan juga pada amplitudo pada sinyal output.

Secara matematis, apabila sinyal input  $x[n]$  menghasilkan output  $y[n]$ , maka input  $kx[n]$  juga menghasilkan  $ky[n]$ .

*IF*



*THEN*

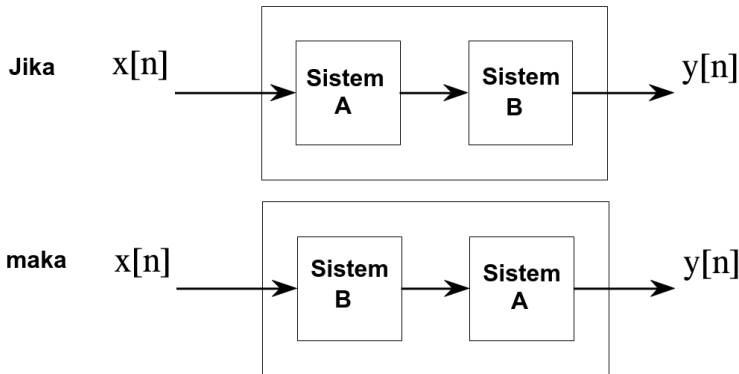


Apabila ada sebuah sinyal input  $x_1[n]$  menghasilkan output  $y_1[n]$ , dan ada sinyal lain  $x_2[n]$  yang menghasilkan  $y_2[n]$ , maka bisa dikatakan sebagai aditif jika input  $x_1[n] + x_2[n]$  menghasilkan output  $y_1[n] + y_2[n]$ .

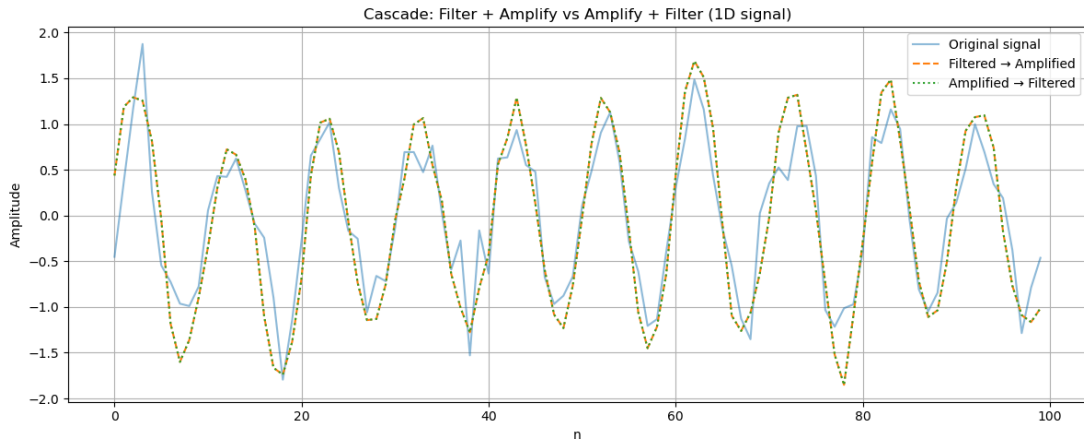
# Sifat Komutatif

Sistem linier juga bersifat komutatif, yang artinya merupakan gabungan lebih dari dua sistem.

Coba dibayangkan dua sistem yang dikombinasikan di mana output dari sistem satu menjadi input ke sistem lainnya (*cascade*). Apabila setiap sistem linier, maka kombinasi dari keseluruhannya adalah linier.



# Contoh Hasil Cascade pada sinyal 1D



# Contoh Hasil Cascade pada citra 2D

Original



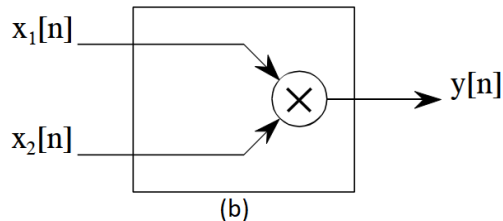
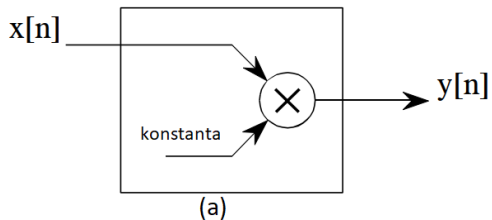
Blur → Amplify



Amplify → Blur



# Linier vs Non-Linier



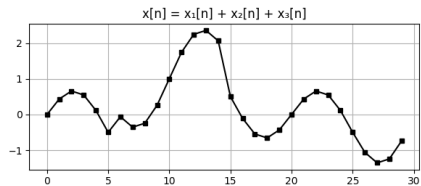
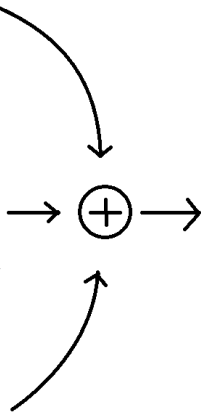
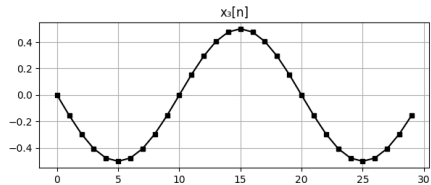
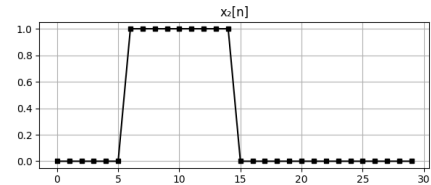
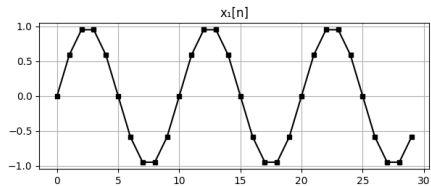
- Perkalian dari dua sinyal, bisa berarti linier atau non-linier
- Pada (a), perkalian antara sinyal dengan konstanta adalah linier
- Pada (b), perkalian tersebut menghasilkan non-linier.



Satu-satunya cara, dalam sistem linier, untuk menggabung beberapa sinyal adalah dengan scaling dan diikuti dengan penjumlahan (*addition*).

Berikut ini contoh penjumlahan tiga buah sinyal  $x_1[n]$ ,  $x_2[n]$ , dan  $x_3[n]$  yang menghasilkan sinyal  $x[n]$ . Proses penggabungan 3 sinyal ini disebut sebagai **sintesis**.

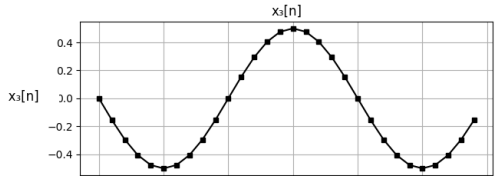
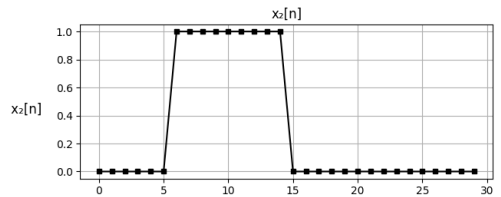
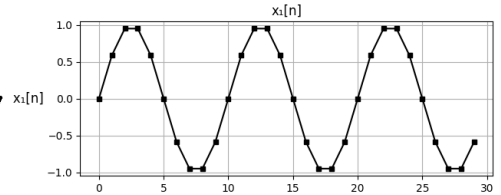
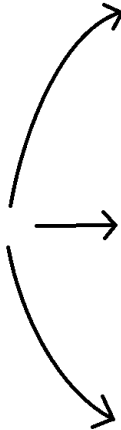
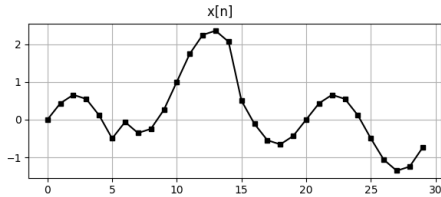
# Sintesis cont'd



**Dekomposisi** adalah kebalikan (*inverse*) dari sintesis, di mana sebuah sinyal dipecah menjadi dua atau lebih komponen. Sebenarnya, proses ini tidak sederhana karena ada kemungkinan yang tidak terhingga dari dekomposisi.

Sebagai contoh sederhana, kita memiliki angka 5 dan 15 yang bisa dijumlahkan menjadi 20. Namun, angka 20 bisa didekomposisi menjadi kombinasi seperti  $1 + 19$ ,  $2+18$ ,  $3+17$ ,  $50-30$ , dst.

# Dekomposisi cont'd



# The Important part

Sekarang kita masuk ke inti dari Pengolahan Sinyal Digital (DSP), yaitu prinsip superposisi, strategi utama untuk menganalisis sinyal dan sistem. Bayangkan ada sebuah sinyal masukan, misalnya  $x[n]$ , yang melewati suatu sistem linier dan menghasilkan sinyal keluaran  $y[n]$ .

Sinyal masukan  $x[n]$  bisa dipecah menjadi beberapa sinyal yang lebih sederhana, misalnya  $x_0[n]$ ,  $x_1[n]$ ,  $x_2[n]$ , dan seterusnya. Sinyal-sinyal sederhana ini disebut komponen sinyal masukan. Masing-masing komponen ini kemudian dimasukkan satu per satu ke dalam sistem, dan menghasilkan keluaran masing-masing:  $y_0[n]$ ,  $y_1[n]$ ,  $y_2[n]$ , dst. Selanjutnya, semua sinyal keluaran ini digabungkan kembali (disintesis) untuk membentuk sinyal keluaran akhir  $y[n]$ .

## The Important part cont'd

1. Hasil keluaran ini akan sama persis dengan jika kita langsung memasukkan sinyal asli  $x[n]$  ke dalam sistem.
2. Untuk memahami bagaimana sistem bekerja terhadap sinyal yang rumit, kita cukup tahu bagaimana sistem tersebut merespons sinyal-sinyal sederhana.
3. Dalam istilah PSD, sinyal input dan output dianggap sebagai penjumlahan (superposisi) dari gelombang-gelombang yang lebih sederhana.

Nah, itulah **superposisi**: memecah hal kompleks jadi bagian-bagian sederhana, mengolah satu per satu, lalu menggabungkannya kembali.

# References



Oppenheim, A. V. (1967).

Generalized superposition.

*Information and Control*, 11(5):528–536.



Sayood, K. (2012).

Mathematical Preliminaries for Transforms, Subbands, and Wavelets.

In Sayood, K., editor, *Introduction to Data Compression (Fourth Edition)*, The Morgan Kaufmann Series in Multimedia Information and Systems, pages 373–407. Morgan Kaufmann, Boston, fourth edition edition.



Smith, S. W. (1997).

*The scientist and engineers' guide to digital signal processing.*

California Technical Publishing, USA.

# The End