

BAB II

TEORI PENUNJANG

2.1 Penginderaan Kompresif

Penginderaan Kompresif atau disebut juga dengan *Compressive Sensing* adalah suatu teknik pengambilan data baru dalam pengolahan sinyal. Hal yang mendasari metode penginderaan kompresif adalah prinsip ketidakpastian yang menyatakan bahwa representasi sinyal dalam kawasan waktu dan frekuensi tidak mungkin bersifat jarang (*sparse*) sekaligus [4]. Sinyal yang jarang diperoleh dari sampel yang sangat sedikit. Manfaat dari penggunaan penginderaan kompresif adalah berkurangnya *bandwidth* untuk transmisi data. Penginderaan kompresif dapat digunakan untuk merekonstruksi sinyal jarang yang dapat dikompresi dari sejumlah pengukuran linier tanpa harus memiliki pengetahuan tentang struktur sinyal yang merupakan poin utama dalam algoritma penginderaan kompresif. Teori penginderaan kompresif diperkenalkan oleh Candes, Tao dan Donoho pada tahun 2004 [5].

Penginderaan Kompresif didasarkan atas dua prinsip, yaitu; *Sparsity* dan *Incoherence*. *Sparsity* adalah sinyal alami seperti suara, gambar atau seismic data dapat disimpan dalam bentuk terkompresi. Ketika basis dipilih dengan benar, besar jumlah koefisien proyeksi nol atau cukup kecil untuk diabaikan. Jika sinyal hanya memiliki koefisien bukan nol dalam jumlah yang terbatas, itu dikatakan jarang. Jika sejumlah besar koefisien proyeksi cukup kecil untuk diabaikan, maka sinyal dikatakan dapat dikompresi.

Incoherence adalah Koherensi mengukur korelasi maksimum antara dua elemen dari dua matriks yang berbeda. Dua matriks ini mungkin mewakili dua basis atau domain yang berbeda berkaitan dengan asas ketidakpastian antara waktu dan frekuensi, yaitu sinyal atau fungsi (waktu) kontinyu tidak mungkin terlokalisasi dengan baik dalam kawasan waktu dan frekuensi secara bersama-sama. Jika ψ adalah sebuah matriks $n \times n$ dengan Ψ_1, \dots, Ψ_n sebagai kolom dan ϕ adalah sebuah matriks $m \times n$ dengan Φ_1, \dots, Φ_n sebagai baris. Maka koheren μ di tentukan sebagai

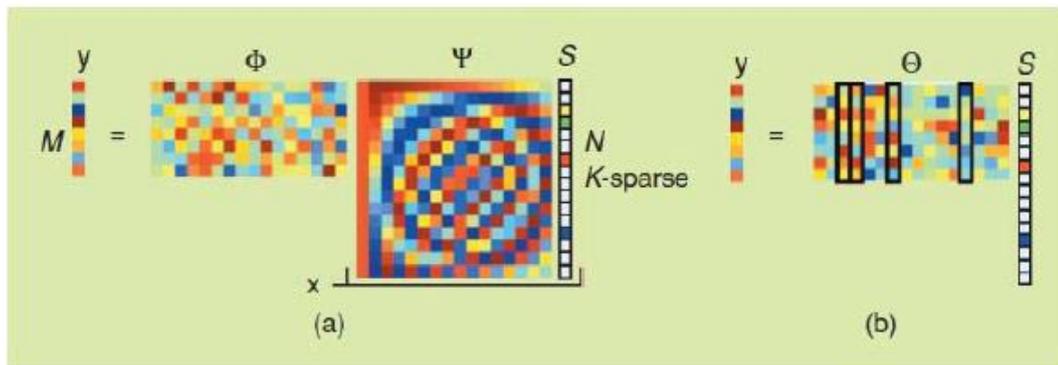
$$\mu(\Phi, \Psi) = \sqrt{n} \max_k |\Psi_k, \Phi_j| \quad (1)$$

untuk $1 \leq j \leq n$ dan $1 \leq k \leq m$ ini mengikuti dari aljabar linear, yaitu:

$$\leq \mu(\Phi, \Psi) \leq \sqrt{n} \quad (2)$$

2.1.1 Sensing Matrix pada Penginderaan Kompresif

Sensing Matrix atau *Measurement Matrix* merupakan matriks yang diperoleh dari hasil pengukuran. Pada proses dari penginderaan kompresif sangat bergantung pada dua aspek, yaitu; *sensing matrix* Φ dan *dictionary/basic* Ψ yang menangkap struktur ringkas dari suatu sinyal [6]. Salah satu faktor penting dari proses penginderaan kompresif adalah untuk memilih *sensing matrix* Φ yang tepat. Matriks penginderaan jarang Φ terdiri dari sejumlah kecil (k) elemen non-nol yang ditarik dari *independent identically distributed* (iid) Gaussian atau Bernoulli variabel secara acak maka dimungkinkan untuk memulihkan sinyal k -sparse dari panjang N , dari jumlah pengukuran M yang lebih sedikit, menggunakan program linier. *Sensing matrix* pada penginderaan kompresif yang dihasilkan merupakan suatu matriks yang memiliki sedikit nilai bukan nol dari suatu citra yang akan digunakan untuk merekonstruksi kembali citra tersebut.



Gambar 2.1 Persamaan *Matrix Sensing*.

$$y = \Phi s, \text{ dimana } s = \Psi s \quad (3)$$

$$y = \Phi \Psi s, \text{ dimana } A = \Phi \Psi \quad (4)$$

$$y = A s \quad (5)$$

Keterangan :

Φ : Matriks pengukuran acak.

ψ : Matriks transformasi.

A : *Matrix Sensing*.

s : Matriks citra asli.

y : Matriks random dari citra asli.

dimana $A = \Phi\psi$ adalah matriks $m \times n$. Proses pengukuran Φ dilakukan secara random yang bergantung pada sinyal x . Proses penginderaan kompresif di atas diilustrasikan pada Gambar 2.1 [7].

2.1.2 *Signal Sparsity*

Signal Sparsity atau sinyal jarang merupakan sinyal yang memiliki elemen nol lebih banyak dibandingkan dengan elemen bukan nol sebagai penyusun sinyal, jika sinyal s sama sekali tidak memiliki elemen bukan nol maka disebut dengan *sparse*. Jika ditentukan nilai s sebagai nilai vektor maka, s bisa dianggap sebagai derajat kebebasan.

Konsep *sparsity* pada proses penginderaan kompresif adalah pada proses *sensing* yang dilakukan pada nilai yang terkompres sehingga memiliki nilai standar di bawah dari standar *Nyquist-rate*. Dalam proses komputasi konsep *sparsity* sangat penting digunakan karena dapat menyimpan banyak komputasi saat menjalankan beberapa operasi. *Signal Sparsity* ini dapat direkonstruksi walaupun dengan jumlah sampel yang dimiliki tidak mencapai standar *Nyquist-rate* yang sudah banyak digunakan tapi menggunakan metode pencupikan penginderaan kompresif.

Konsep *sparsity* digunakan dalam proses penginderaan kompresif, karena dalam proses CS membutuhkan nilai bukan nol pada suatu sinyal yang dapat digunakan untuk melakukan rekonstruksi suatu sinyal. Menggunakan konsep *Sparsity*, membentuk sinyal menjadi suatu nilai jarang dengan membagi antara nilai nol dan nilai bukan nol pada suatu sinyal, sehingga memudahkan untuk mencari

nilai-nilai penting yang bisa digunakan untuk melakukan proses rekonstruksi [8]. Pada penelitian ini sinyal *Sparsity* didapatkan melalui proses transformasi yang dilakukan pada sinyal citra. Transformasi yang digunakan adalah *Discrete Cosine Transform*.

2.1.3 *Lossy dan Lossless Compression*

Lossy dan *Lossless Compression* adalah dua metode pada pengolahan citra untuk mengurangi ukuran pada suatu citra. Pada kasus *lossy compression*, data dan ukuran dari citra yang dikompresi menggunakan metode *lossy compression* akan berkurang. Namun data yang hilang pada saat dikompresi tidak dapat dikembalikan seperti semula. Sebaliknya pada *lossless compression*, data yang dikompresi harus bisa dikembalikan lagi seperti semula tanpa adanya data yang dihilangkan dari metode kompresi tersebut.

Metode kompresi pada *lossy* dapat mengakibatkan sebagian informasi yang diperlukan dalam proses dapat mengakibatkan kesalahan pada citra tersebut. Sedangkan pada metode *lossless* semua informasi harus bisa dikembalikan lagi seperti semula tanpa adanya kesalahan atau kerusakan pada citra tersebut [9].

2.1.4 *Discrete Cosine Transform*

Discrete Cosine Transform (DCT) pertama kali diusulkan pada tahun 1972 yang ditujukan untuk kompresi citra dan dinyatakan sebagai algoritma yang paling efisien untuk melakukan kompresi citra. Sampai saat ini DCT merupakan teknik transformasi yang paling banyak digunakan pada pemrosesan sinyal dan transformasi linier yang paling banyak digunakan pada proses pengkompresan data [8].

Pada kompresi citra JPEG, citra dibagi menjadi blok 8x8 piksel kemudian 2DDCT diaplikasikan pada tiap-tiap blok 8x8. Pada dekompresi citra, IDCT diaplikasikan pada tiap-tiap blok 8x8 [10]. Persamaan *Discrete Cosine Transform* dapat dilihat pada persamaan berikut:

$$F(u, v) = a_j a_i \sum_{j=0}^{M-1} \sum_{i=0}^{N-1} A(mn) \cdot \cos \left[\frac{\pi \cdot u}{2 \cdot M} (2j + 1) \right] \cos \left[\frac{\pi \cdot v}{2 \cdot N} (2i + 1) \right] \quad (6)$$

dimana,

$$a_i = \begin{cases} \frac{1}{\sqrt{n}}, & \text{jika } i = 0 \\ \sqrt{\frac{2}{N}} \cos \frac{(2j+1)i\pi}{2N}, & \text{jika } i \neq 0 \end{cases} \quad (7)$$

dan,

$$a_j = \begin{cases} \frac{1}{\sqrt{n}}, & \text{jika } j = 0 \\ \sqrt{\frac{2}{N}} \cos \frac{(2j+1)i\pi}{2N}, & \text{jika } j \neq 0 \end{cases} \quad (8)$$

F(0,0)	F(0,1)	F(0,2)	F(0,3)	F(0,4)	F(0,5)	F(0,6)	F(0,7)
F(1,0)	F(1,1)	F(1,2)	F(1,3)	F(1,4)	F(1,5)	F(1,6)	F(1,7)
F(2,0)	F(2,1)	F(2,2)	F(2,3)	F(2,4)	F(2,5)	F(2,6)	F(2,7)
F(3,0)	F(3,1)	F(3,2)	F(3,3)	F(3,4)	F(3,5)	F(3,6)	F(3,7)
F(4,0)	F(4,1)	F(4,2)	F(4,3)	F(4,4)	F(4,5)	F(4,6)	F(4,7)
F(5,0)	F(5,1)	F(5,2)	F(5,3)	F(5,4)	F(5,5)	F(5,6)	F(5,7)
F(6,0)	F(6,1)	F(6,2)	F(6,3)	F(6,4)	F(6,5)	F(6,6)	F(6,7)
F(7,0)	F(7,1)	F(7,2)	F(7,3)	F(7,4)	F(7,5)	F(7,6)	F(7,7)

Gambar 2.2 Koefisien DCT

2.1.5 *Inverse Discrete Cosine Transform*

Citra yang sudah ditransformasi berada pada domain frekuensi, untuk membentuk kembali citra dengan domain spasial maka perlu dilakukan *inverse* DCT. Persamaan yang digunakan untuk melakukan *inverse* DCT adalah:

$$A(m, n) = a_j a_i \sum_{j=0}^{M-1} \sum_{i=0}^{N-1} F(u, v) \cdot \cos \left[\frac{\pi \cdot u}{2 \cdot M} (2j + 1) \right] \cos \left[\frac{\pi \cdot v}{2 \cdot N} (2i + 1) \right] \quad (9)$$

Secara matematika, IDCT benar-benar merupakan kebalikan dari persamaan DCT dan tidak akan menghilangkan satupun koefisien.

2.1.6 *Minimum ℓ_1 -normal reconstruction*

Algoritma rekonstruksi digunakan untuk merekonstruksi nilai *sparse* dari hasil penginderaan kompresif. *Minimum ℓ_1 -normal reconstruction* merupakan salah satu algoritma rekonstruksi yang bisa digunakan. Algoritma rekonstruksi merupakan hasil dari pengukuran nilai M pada pengukuran vektor y dari pengukuran matriks acak Φ . Basis ψ dan rekonstruksi sinyal x dengan panjang N atau koefisien *sparse* s. Algoritma *minimum ℓ_1 -norm reconstruction* [3] adalah:

$$s = \operatorname{argmin} \|x\|_1 \quad \text{untuk} \quad \Phi s = y \quad (10)$$

2.2 **Peak Signal-to-Noise Ratio (PSNR)**

Peak Signal-to-Noise Rasio (PSNR) merupakan suatu rasio tingkat kemiripan citra yang biasa digunakan dalam proses kompresi citra untuk menentukan kualitas pada hasil rekonstruksi citra [11]. Parameter PSNR digunakan untuk melihat tingkat kemiripan citra antara citra asli dengan citra terekonstruksi. PSNR dapat didefinisikan melalui nilai *Mean Square Error* (MSE) atau nilai rata-rata kuadrat dari kesalahan atau *error*. MSE tidak memiliki nilai satuan sedangkan PSNR ditulis dengan satuan desibel (dB), sehingga dua gambar memiliki tingkat kemiripan yang rendah jika nilai PSNR dibawah 30dB. Semakin besar nilai PSNR maka kedua buah gambar memiliki tingkat kemiripan yang tinggi. Persamaan berikut merupakan persamaan untuk menghitung nilai MSE.

$$MSE = \frac{1}{MN} \sum_0^{m-1} \sum_0^{n-1} ||f(i,j) - g(i,j)||^2 \quad (11)$$

dimana,

M = merupakan baris dari piksel dan i merupakan indeks baris.

N = merupakan kolom dari piksel dan j merupakan indeks kolom.

f = merupakan data matriks sebelum rekonstruksi.

g = merupakan data matriks sesudah rekonstruksi.

$$PSNR = 10 \log_{10} \left(\frac{MAX_1^2}{MSE} \right) \quad (12)$$

dimana, *MAX* merupakan nilai piksel maksimum pada suatu citra, untuk citra 8-bit nilai *MAX* adalah 255.

2.3 Raspberry Pi

Raspberry Pi adalah sebuah *single purpose* komputer yang berukuran sebesar kartu kredit. Raspberry Pi pertama kali diluncurkan pada Februari 2012, tujuannya adalah untuk menciptakan dan membangkitkan minat *programmer* dalam bidang komputasi. Pemilihan Raspberry Pi pada tugas akhir ini dikarenakan Raspberry Pi lebih kompleks untuk multimedia dan berbasis dasar linux yang banyak dikembangkan oleh siapapun yang bersifat *open source*.

Raspberry Pi menggunakan sistem on a chip (SoC) dari Broadcom BCM2835 dan tidak menggunakan hard disk, namun menggunakan SD Card untuk proses booting dan penyimpanan data jangka panjang. diinginkan. Jadi mikrokontroler bertugas sebagai otak yang mengendalikan input, proses, dan output pada sebuah rangkaian elektronik. Raspberry Pi ini berjalan diatas sistem operasi yang sifatnya open source seperti Raspbian Wheezy, Pidora, Risc OS, Rasp BMC, Arch Linux, dan lain-lain. Untuk bahasa pemrogramannya menggunakan bahasa pemrograman Python yang cocok untuk digunakan pada penelitian ini [12]