ABSTRAK

PEMAMPATAN CITRA MENGGUNAKAN HAAR WAVELET TRANSFORMASI

John Binsar Tua Simalango
(10102394)
Fakultas Teknik Informatika
Universitas Komputer Indonesia

1. PENDAHULUAN

Dewasa ini proses komunikasi dalam bertukar informasi tidak hanya menggunakan tulisan berupa teks saja namun juga menggunakan citra sebagai pelengkap informasi sehingga informasi yang akan disampaikan dapat dimengerti lebih baik lagi.

Citra yang baik sangat mendukung terhadap informasi yang akan disampaikan, jika citra yang ditampilkan kurang baik maka hal itu dapat mempengaruhi terhadap informasi yang akan disampaikan bisa tidak sesuai apa yang diharapkan sebelumnya.

Besarnya ukuran citra dapat berpengaruh dalam proses pengiriman dan penerimaan citra khususnya bila citra tersebut dikirim atau diterima melalui internet, semakin besar ukuran citra maka akan semakin lambat proses pengiriman atau penerimaannya dan memerlukan tempat yang cukup banyak untuk menyimpan citra yang besar tersebut.

2. PENYIMPANAN CITRA DIGITAL

Citra digital membutuhkan kapasitas memori yang cukup besar untuk disimpan, dan ketika kita mengambilnya di Internet, bisa menghabiskan waktu yang sangat lama untuk men-*download*. Beberapa teknik digunakan untuk memampatkan citra digital tersebut agar bisa menghemat memori. Salah satunya Haar wavelet transformasi.

File yang berisi citra bisa berukuran sangat besar dan dengan cepat bisa menghabiskan ruang pada hard disk. Suatu citra grayscale (abu-abu) yang berukuran 256x256 pixel misalnya membutuhkan 65536 elemen untuk disimpan. Sedangkan untuk citra biasa (berwarna) yang berukuran 640x480 akan membutuhkan jutaan elemen! Ukuran dari file ini juga dapat menyebabkan proses download dari internet bertambah lama.

Haar wavelet transformasi dapat melakukan pemampatan citra sehingga dapat menghemat media menyimpanan, dan juga menyebabkan pengiriman lewat internet bisa lebih cepat.

Seperti yang sudah disebutkan sebelumnya, suatu citra grayscale berukuran 256x256 pixel membutuhkan 65536 elemen untuk disimpan. Maksudnya citra tersebut disimpan sebagai matrix berukuran 256x256, yang mana tiap-tiap elemennya merupakan suatu angka yang bernilai dari nol (untuk warna hitam) sampai bilangan positif terbesar (untuk warna putih).

Sedangkan untuk citra 256x256 RGB berwarna membutuhkan tiga buah matrix berukuran 256x256 yang masing-masing untuk warna merah (Red), hijau (Green), dan blue (Biru). Kita bisa menggunakan matrix ini dan beberapa perhitungan aljabar linear untuk memaksimalkan pemampatan sambil mengatur detail citra agar tidak rusak.

2.1 Pemampatan Citra dengan Transformasi Haar Wavelet

Dalam melakukan pemampatan menggunakan transformasi Haar wavelet, dilakukan dengan langkah-langkah berikut:

1. Konversi citra menjadi bentuk matriks (I)
2. Lakukan *averaging and differencing* pada matriks tersebut dengan rumus $T = W^T I W$.

Matriks hasil akan cenderung jarang (*sparse*).

3. Tentukan nilai *epsilon* (ε), dan ubah semua elemen pada | T | yang lebih kecil atau sama dengan ε menjadi nol. Hasilnya akan berupa matriks jarang yang disimpan di S.
4. Buat matriks baru (anggap saja R) dari matriks S dengan menggunakan rumus $R = W S W^{-1}$.

3. TAHAP IMPLEMENTASI

Tahap implementasi sistem merupakan tahap yang dilakukan setelah sistem dianalisis dan dirancang. Dimana pada tahap ini mencakup kegiatan memindahkan hasil analisis dan rancangan ke dalam logika pemograman sesuai dengan bahasa pemograman yang dipilih. Pada tahap ini, semua kelas akan disebut sebagai objek.

3.1 Tahap I

Tahap I merupakan versi "pembuka" dari aplikasi yang akan dibuat. Pada tahap ini, HaaVe hanya berfungsi untuk membuka dan menampilkan file citra .BMP saja. Dengan cara tersebut kita tinggal mengembangkan aplikasi ke tahap berikutnya dengan mudah tanpa harus merekonstruksi *source code* program.

Sistem HaaVe memiliki tampilan jenis MDI (*Multiple Document Interface*) yang diimplementasikan sebagai objek MDIMainForm yang memiliki *child* FBMP. FBMP ini memiliki tugas untuk menampilkan file .BMP yang dibuka oleh *user*. Setelah dilakukan operasi pembukaan file, *property* BMPsize akan diinisialisasikan.

3.2 Tahap II

Tahap II merupakan tahap inti dari sistem HaaVe. Pada tahap ini sudah diimplementasikannya fungsi untuk memampatkan citra dengan menambah objek baru, yaitu FWHL. Namun, objek FHWL hanya berfungsi sebagai penerima citra hasil pemampatan saja. Yang melakukan proses pemampatan adalah objek FBMP, karena pada objek inilah terdapat informasi-informasi tentang citra .BMP yang diperlukan untuk melakukan proses pemampatan.

Setelah memasuki tahap ini, proses inisialisasi setelah proses pembukaan file citra tidak hanya dilakukan pada *property* BMPsize, tetapi juga dilakukan pada *property* Epsilon dan Bytesize. Nilai *property* Epsilon ditetapkan 25, sedangkan *property* Bytesize yang menyatakan ukuran dari matriks diisi dengan rumus "lebar_citra x tinggi_citra" (satuan lebar dan tinggi adalah pixel).

Property Zero dan NonZero akan diisi setelah proses pemampatan berhasil dilakukan. Setelah dipampatkan, hasilnya dikirim ke objek FHWL untuk kemudian ditampilkan sebagai citra baru. Hasilnya ditampilkan pada tampilan (*window*) yang berbeda agar *user* bisa membandingkan citra asal dengan citra hasil pemampatan.

3.3 Tahap Akhir

Pada tahap ini hanya dilakukan penambahan fungsi penyimpanan citra hasil pemampatan agar bisa dibuka kembali sewaktu-waktu

--ooOOOoo--