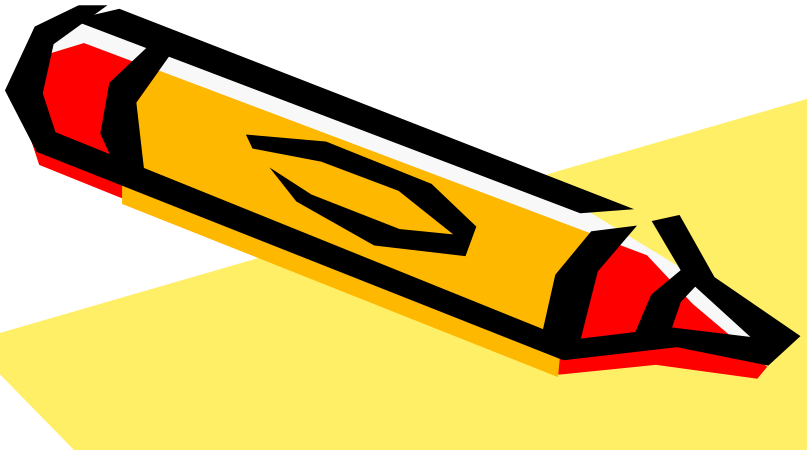
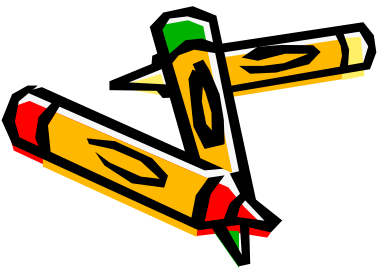


Digital Watermarking

Bahan Kuliah IF4020 Kriptografi

Oleh: Rinaldi Munir



Pengantar

Citra (image) atau Gambar

”Sebuah gambar bermakna lebih dari seribu kata”

(*A picture is more than a thousand words*)

Termasuk gambar-gambar animasi ini

Fakta

- Jutaan gambar/citra digital bertebaran di internet via *email*, *website*, *bluetooth*, dsb

- Siapapun bisa mengunduh citra dari web, meng-*copy*-nya, menyunting, mengirim, memanipulasi, dsb.

- Memungkinkan terjadi pelanggaran HAKI:
  - mengklaim citra orang lain sebagai milik sendiri (pelanggaran kepemilikan)
  - meng-*copy* dan menyebarkan citra tanpa izin pemilik (pelanggaran *copyright*)
  - mengubah konten citra sehingga keasliannya hilang

Kasus 1: Alice dan Bob sama-sama mengklaim gambar ini miliknya

Siapa pemilik gambar ini sesungguhnya? Hakim perlu memutuskan!

Kasus 2: Alice memiliki sebuah gambar UFO hasil jepretannya. Bob menggandakan dan menyebarkannya tanpa izin dari Alice

Kasus 3: Alice memiliki sebuah gambar hasil fotografi. Bob memodifikasi gambar tersebut dengan menggunakan Photoshop

Mana gambar yang asli?

Original

Hasil pengubahan

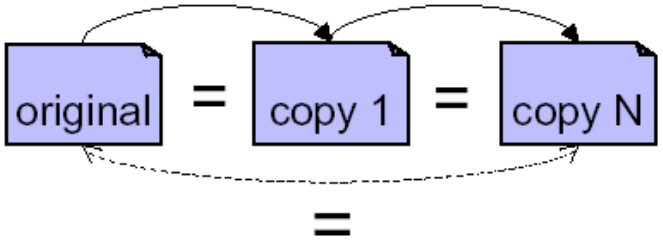
(a) Clinton and Monica

Foto mana yang asli?

(b) Clinton and Hillary

Semua kasus-kasus di atas karena karakteristik (kelebihan sekaligus kelemahan) dokumen digital adalah:

- Tepat sama kalau digandakan

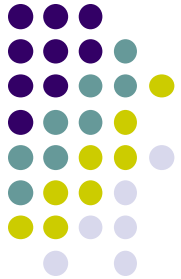


- Mudah didistribusikan (misal: via internet)
- Mudah di-edit (diubah) dengan *software*

Tidak ada perlindungan terhadap citra digital!!!!

Solusi untuk masalah perlindungan citra di atas adalah:

Image Watermarking!!!!!!

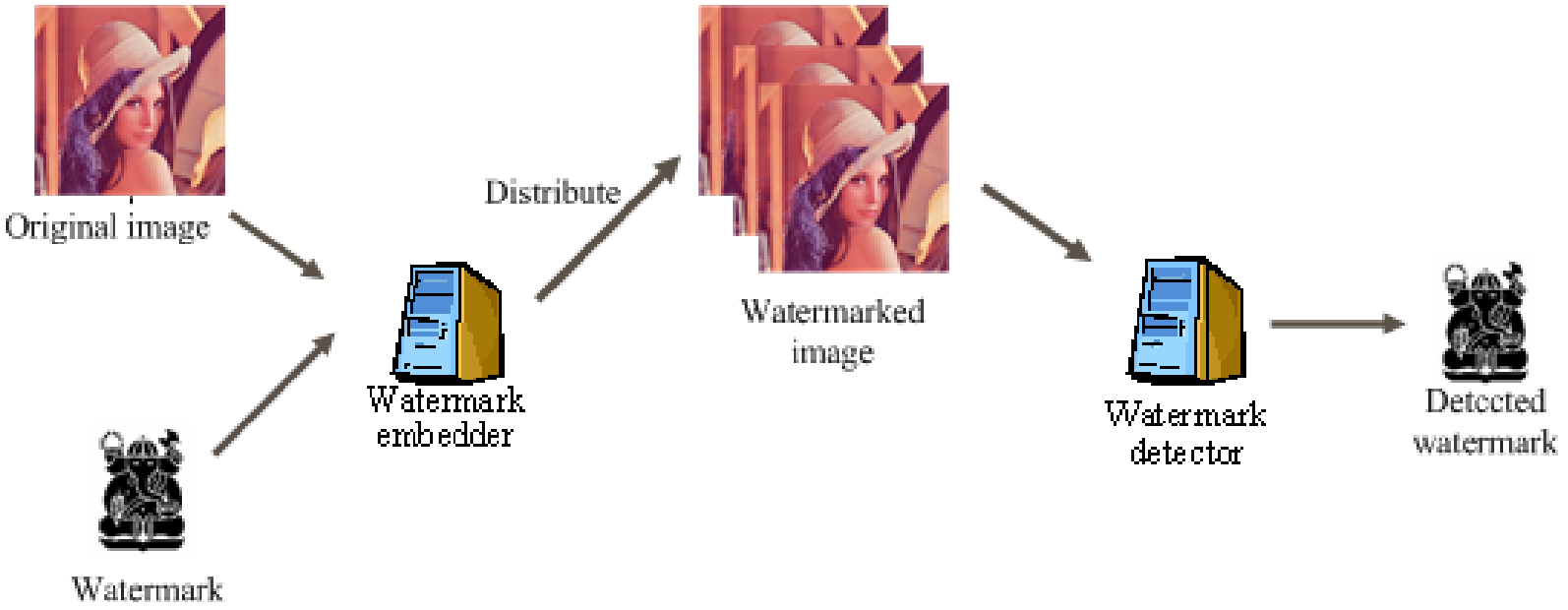


Digital Watermarking

Image Watermarking

- *Image Watermarking*: penyisipan informasi (*watermark*) yang mengacu pada pemilik gambar untuk tujuan melindungi kepemilikan, *copyright* atau menjaga keaslian konten
- *Watermark*: teks, gambar logo, audio, data biner (+1/-1), barisan bilangan riil
- Penyisipan *watermark* ke dalam citra sedemikian sehingga tidak merusak kualitas citra.

Model Image Watermarking



- *Watermark* melekat di dalam citra
- Penyisipan *watermark* tidak merusak kualitas citra
- *Watermark* dapat dideteksi/ekstraksi kembali sebagai bukti kepemilikan/*copyright* atau bukti adanya modifikasi

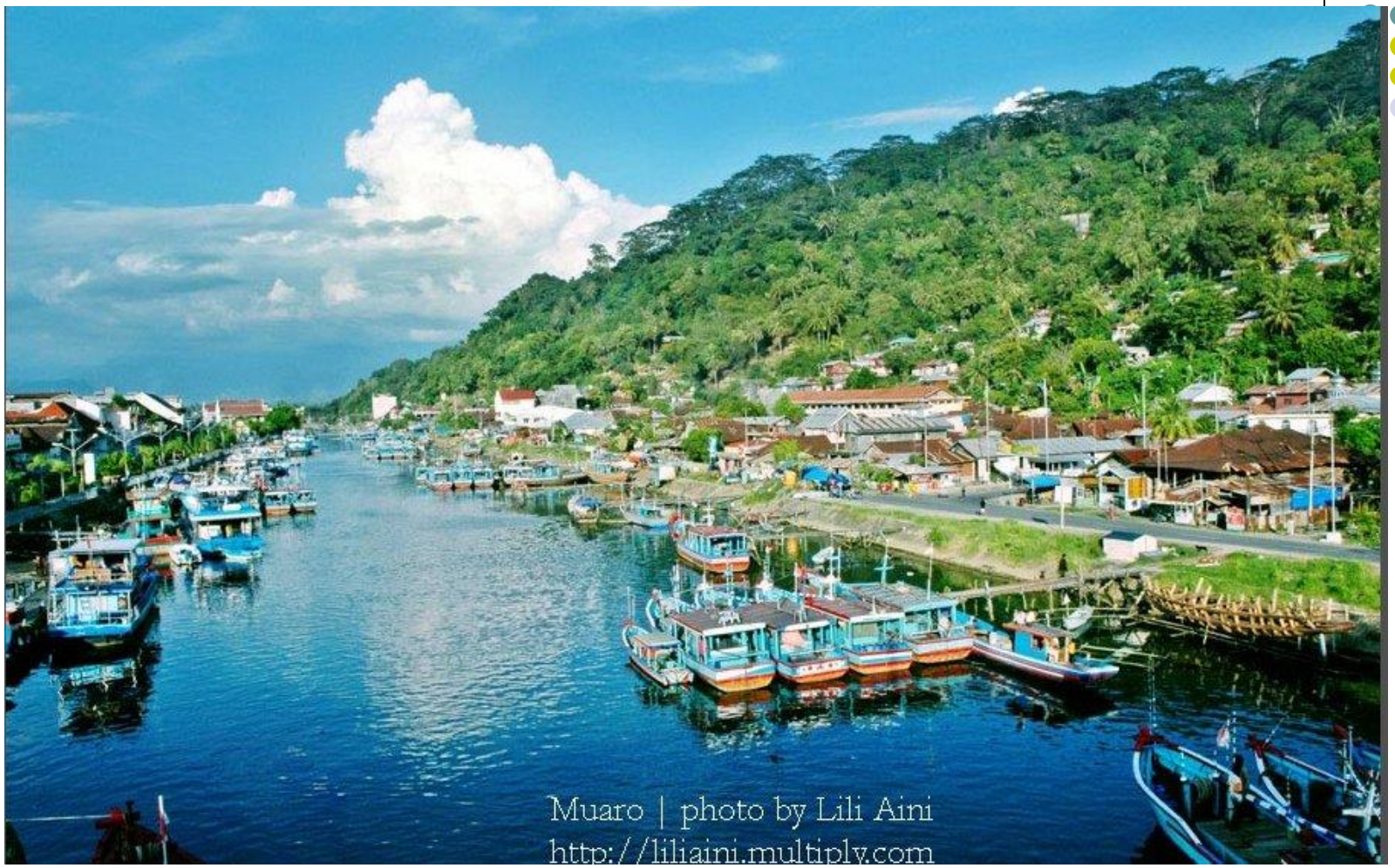
Cara-cara Konvensional Memberi Label *Copyright*

- Label *copyright* ditempelkan pada gambar.

- Kelemahan: tidak efektif melindungi *copyright* sebab label bisa dipotong atau dibuang dengan program pengolahan citra komersil (ex: *Adobe Photoshop*).

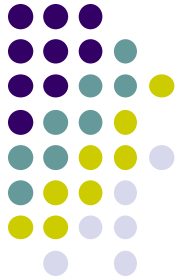
Original image + label *copyright*

Cropped image



Label kepemilikan

Dengan teknik *watermarking…*



- *Watermark* disisipkan ke dalam citra digital.
- *Watermark* terintegrasi di dalam citra digital
- Kelebihan:
  1. Penyisipan *watermark* tidak merusak kualitas citra, citra yang diberi *watermark* terlihat seperti aslinya.
  2. Setiap penggandaan (*copy*) data multimedia akan membawa *watermark* di dalamnya.
  3. *Watermark* tidak bisa dihapus atau dibuang
  4. *Watermark* dapat dideteksi/ekstraksi kembali sebagai bukti kepemilikan /*copyright* atau deteksi perubahan

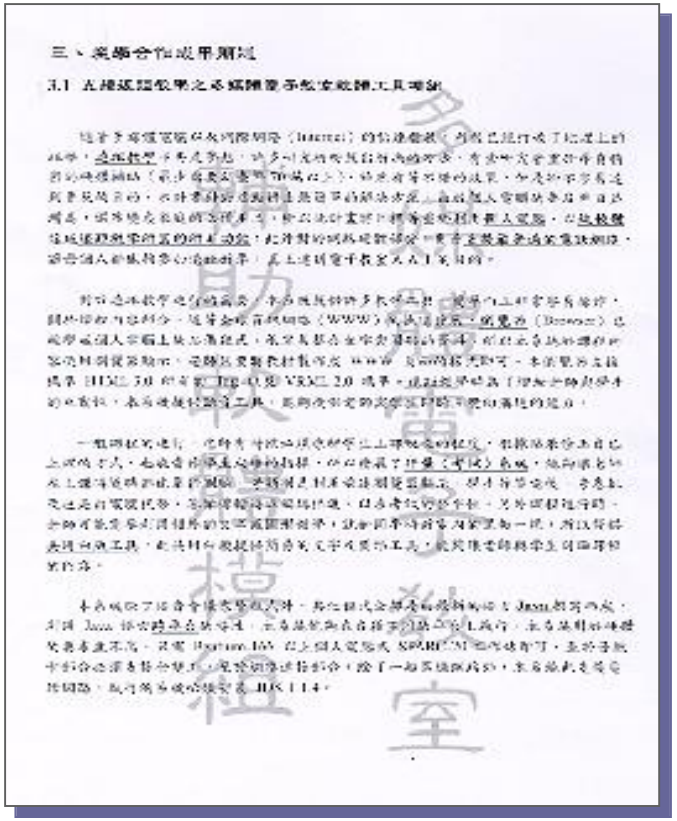
Sejarah *Watermarking*

- Abad 13, pabrik kertas di Fabriano, Italia, membuat kertas yang diberi *watermark* dengan cara menekan bentuk cetakan gambar pada kertas yang baru setengah jadi.

- Ketika kertas dikeringkan terbentuklah suatu kertas yang ber-*watermark*. Kertas ini biasanya digunakan oleh seniman/sastrawan untuk menulis karya seni.

- Kertas yang sudah dibubuhi tanda-air dijadikan identifikasi bahwa karya seni di atasnya adalah milik mereka.

- Bangsa Cina melakukan hal yang sama pada pencetakan kertas



Klasifikasi *Watermarking*

1. *Paper watermarking*

   Teknik memberikan impresi pada kertas berupa gambar/logo atau teks.

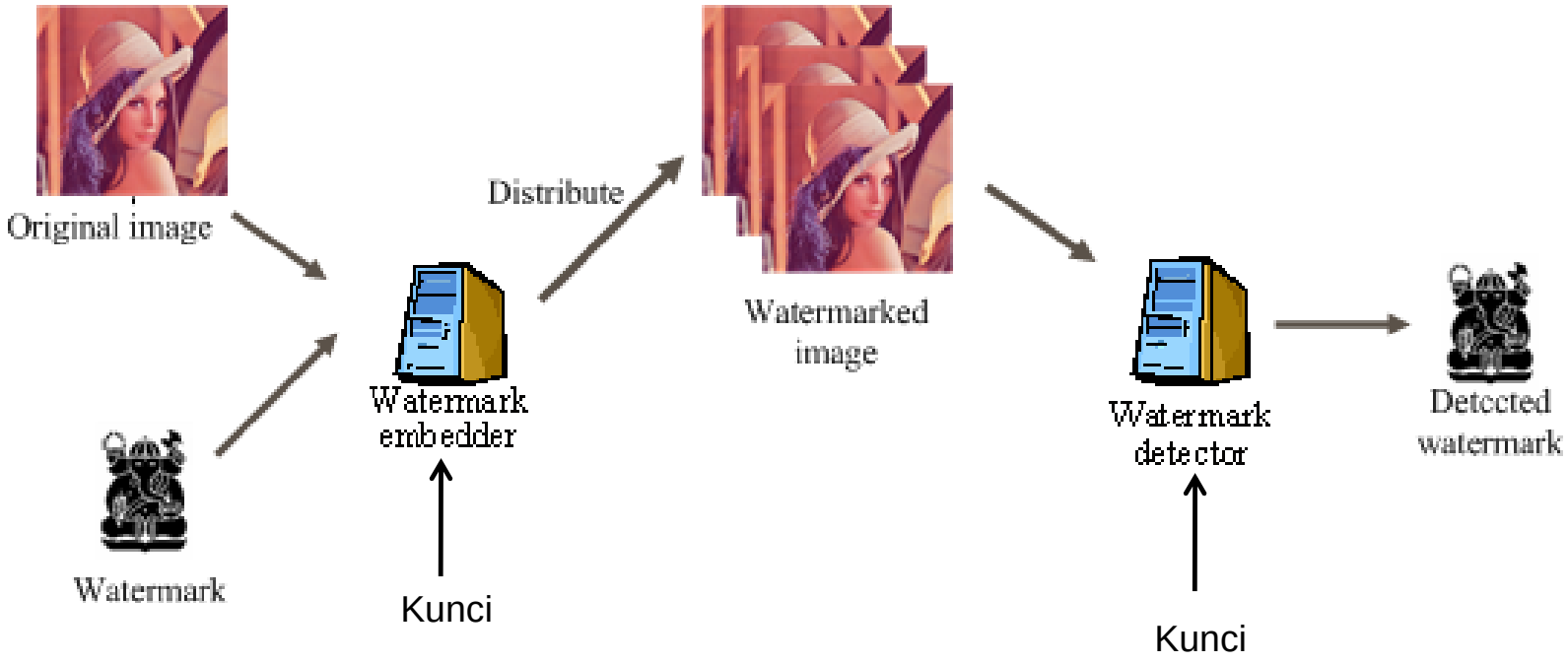
   "Cannot be photocopied or scanned effectively"

   Tujuan: Identifikasi keaslian (otentikasi)

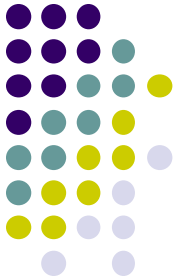
   Digunakan pada: uang, paspor, banknotes ,

2. *Digital Watermarking*

Menyisipkan sinyal digital ke dalam dokumen digital (gambar, audio, video, teks)

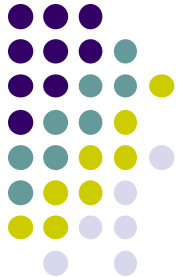


Perbedaan Steganografi dan *Watermarking*



Steganografi:

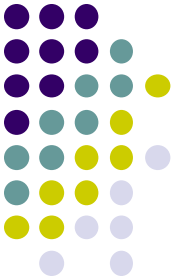
- Tujuan: mengirim pesan rahasia apapun tanpa menimbulkan kecurigaan
- Persyaratan: aman, sulit dideteksi, sebanyak mungkin menampung pesan (*large capacity*)
- Komunikasi: *point-to-point*
- Media penampung tidak punya arti apa-apa (*meaningless*)



***Watermarking*:**

- Tujuan: perlindungan *copyright*, pembuktian kepemilikan (*ownership*), *fingerprinting*
- Persyaratan: *robustness*, sulit dihapus (*remove*)
- Komunikasi: *one-to-many*
- Komentar lain: media penampung justru yang diberi proteksi, *watermark* tidak rahasia, tidak mementingkan kapasitas *watermark*

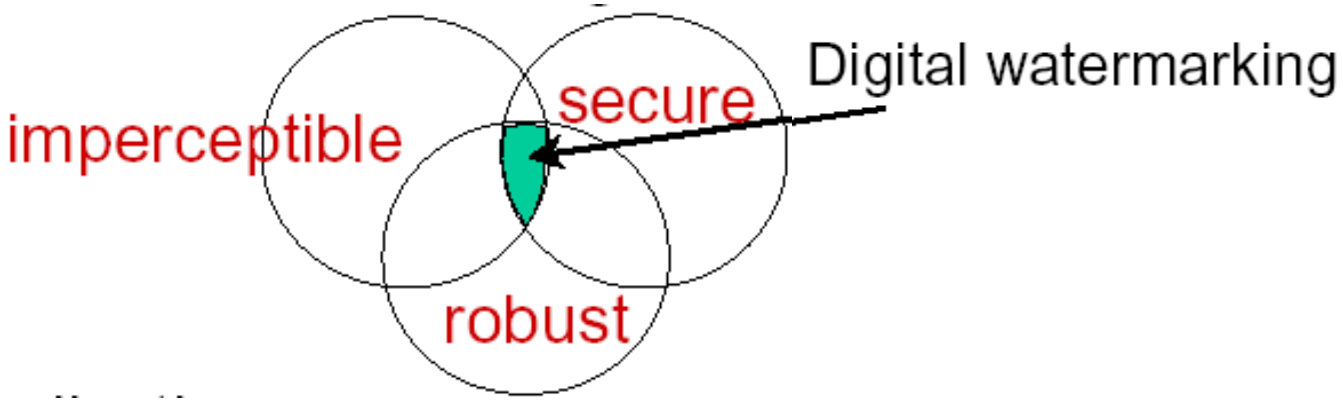
Selain citra, data apa saja yang bisa diberi *watermark*?



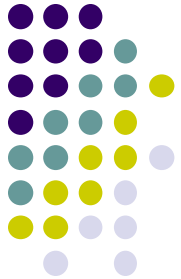
- Citra → *Image Watermarking*
- Video → *Video Watermarking*
- Audio → *Audio Watermarking*
- Teks → *Text Watermarking*
- Perangkat lunak → *Software watermarking*

Image Watermarking

- Persyaratan umum:
  - *imperceptible*
  - *robustness*
  - *secure*



Jenis-jenis *Image Watermarking*



- *Fragile watermarking*

  Tujuan: untuk menjaga integritas/orisinilitas media digital.

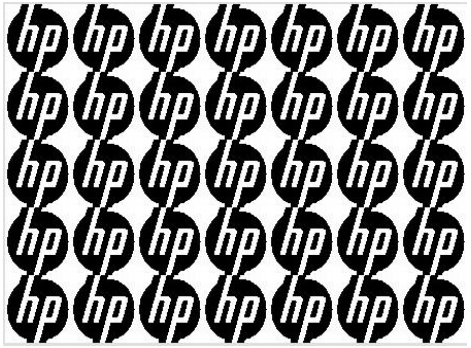
- *Robust watermarking*

  Tujuan: untuk menyisipkan label kepemilikan media digital.

Fragile Watermarking

- *Watermark* rusak atau berubah terhadap manipulasi (*common digital processing*) yang dilakukan pada media.
- Tujuan: pembuktian keaslian dan *tamper proofing*

(a)

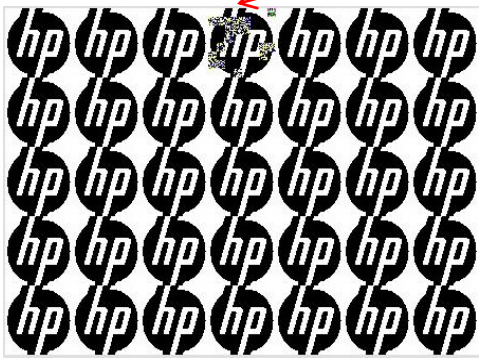


(b)

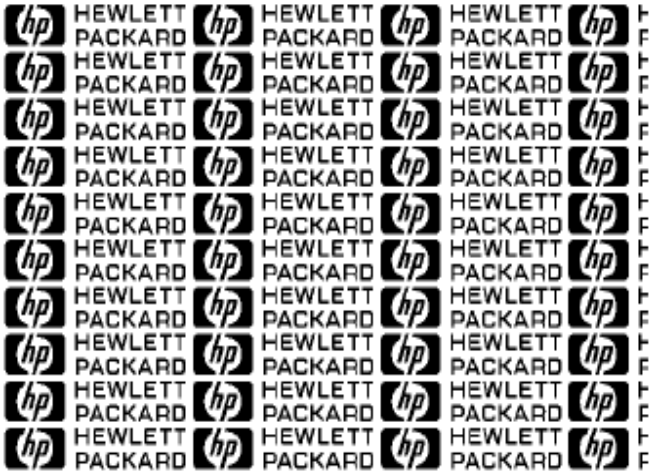
Penambahan *noise*

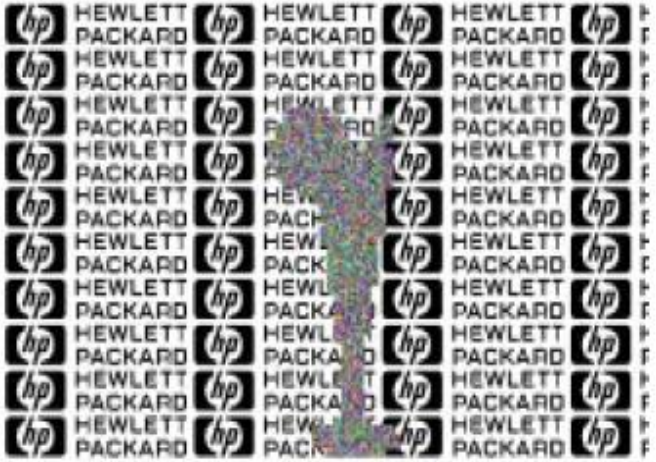
(c)

Watermark rusak



(d)





Contoh *fragile watermarking* lainnya (Wong, 1997)

Bagaimana caranya?

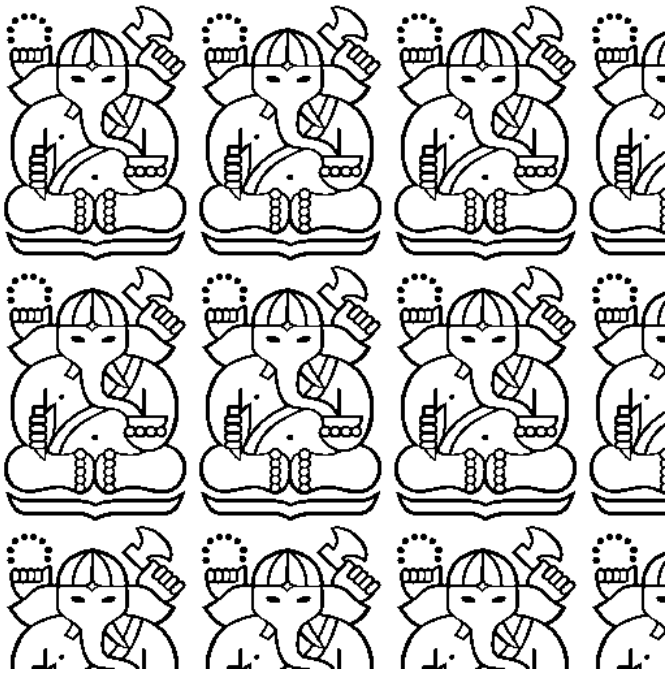
- Pertama, harus mengerti dulu konsep citra digital

- Kedua, mengerti algoritma modifikasi LSB (sudah dijelaskan di dalam materi Steganografi)

Algoritma *Fragile Watermarking*

1. Nyatakan *watermark* seukuran citra yang akan disisipi (lakukan *copy and paste*)

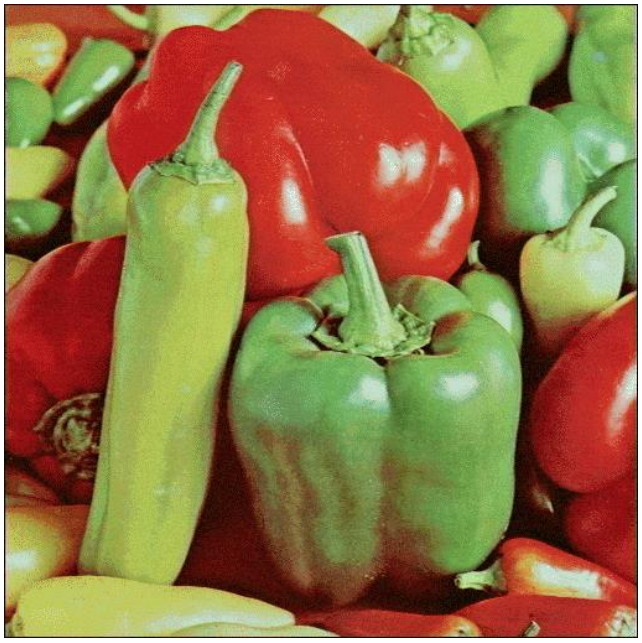
Citra asli



watermark

2. Sisipkan *watermark* pada seluruh *pixel* citra dengan metode modifikasi LSB

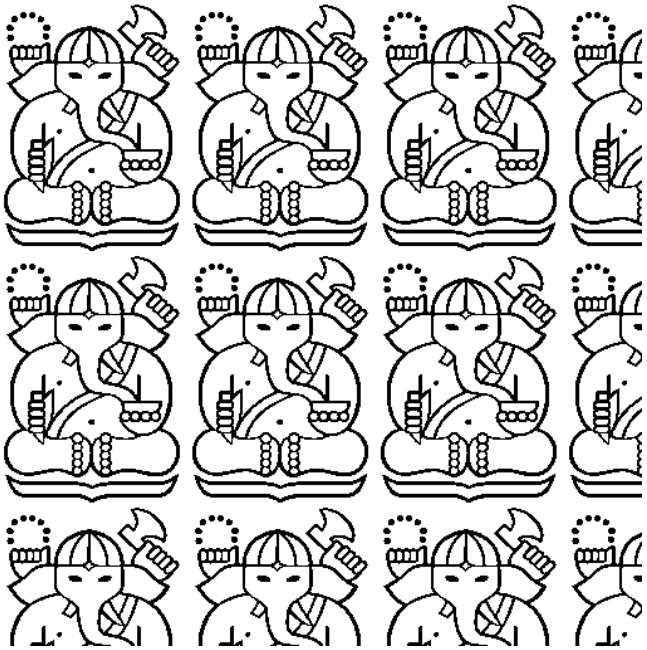
Citra asli



Citra ber-watermark

3. Ekstraksi *watermark* dengan mengambil bit-bit LSB pada setiap *pixel*, lalu satukan menjadi gambar *watermark* semula

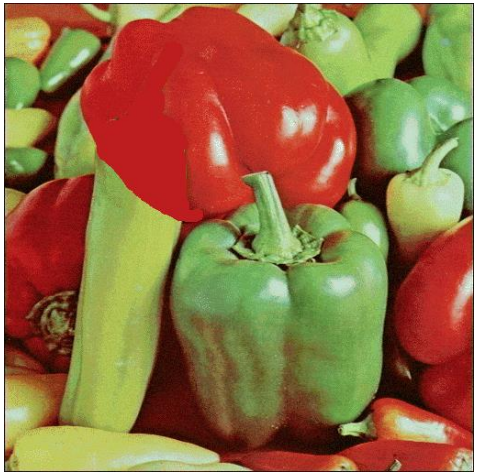
Citra ber-watermark

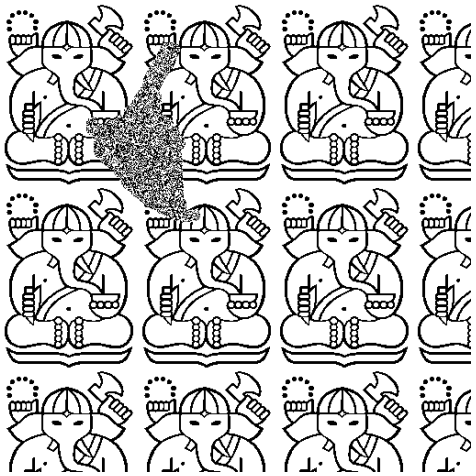


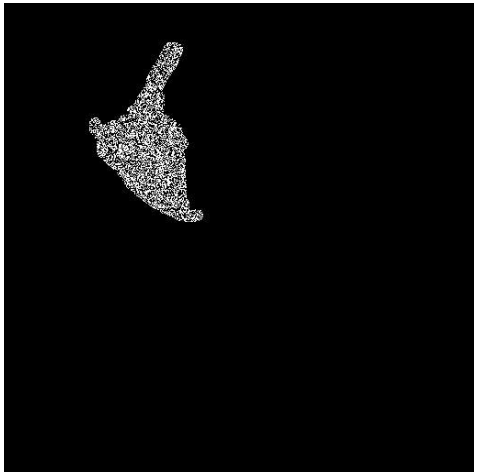
Watermark hasil ekstraksi

Test modifikasi citra ber-*watermark*

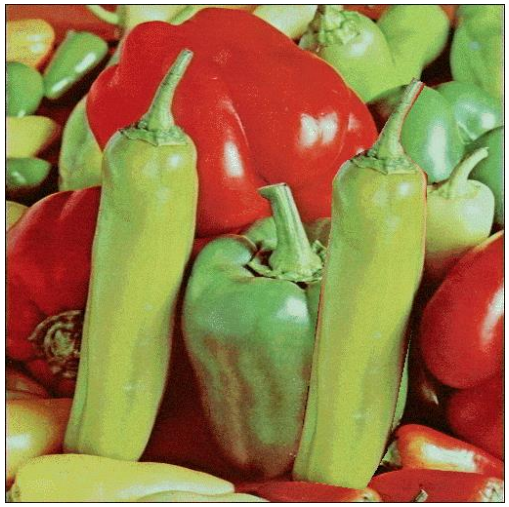
Deletion attack

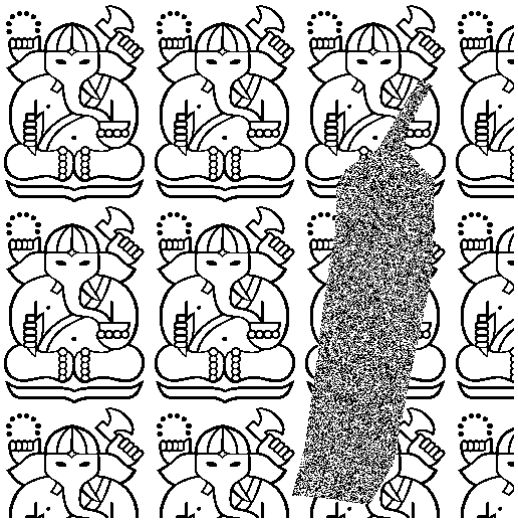


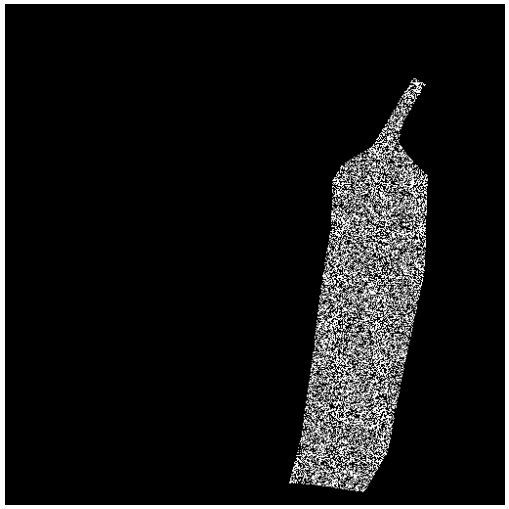


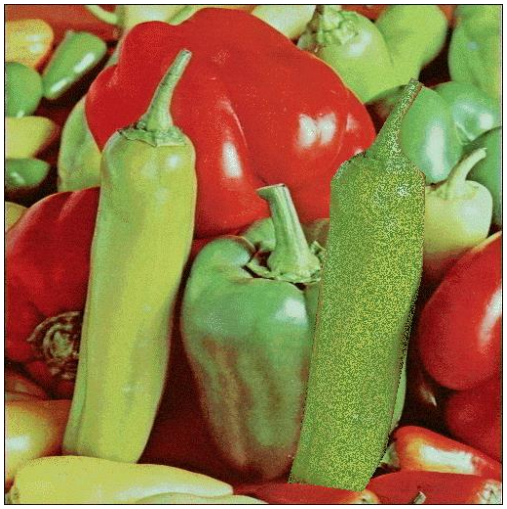


Insertion attack

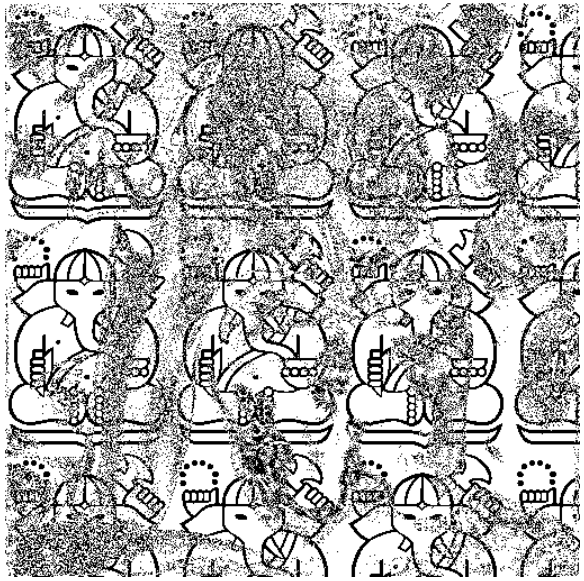








Brightness and contrast attack



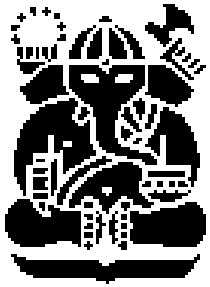
Robust Watermarking

- *Watermark* tetap kokoh (*robust*) terhadap manipulasi (*common digital processing*) yang dilakukan pada media.

  Contoh: kompresi, *cropping*, *editing*, *resizing, dll*
- Tujuan: perlindungan hak kepemilikan dan *copyright*

Original image

Stego-image



watermark

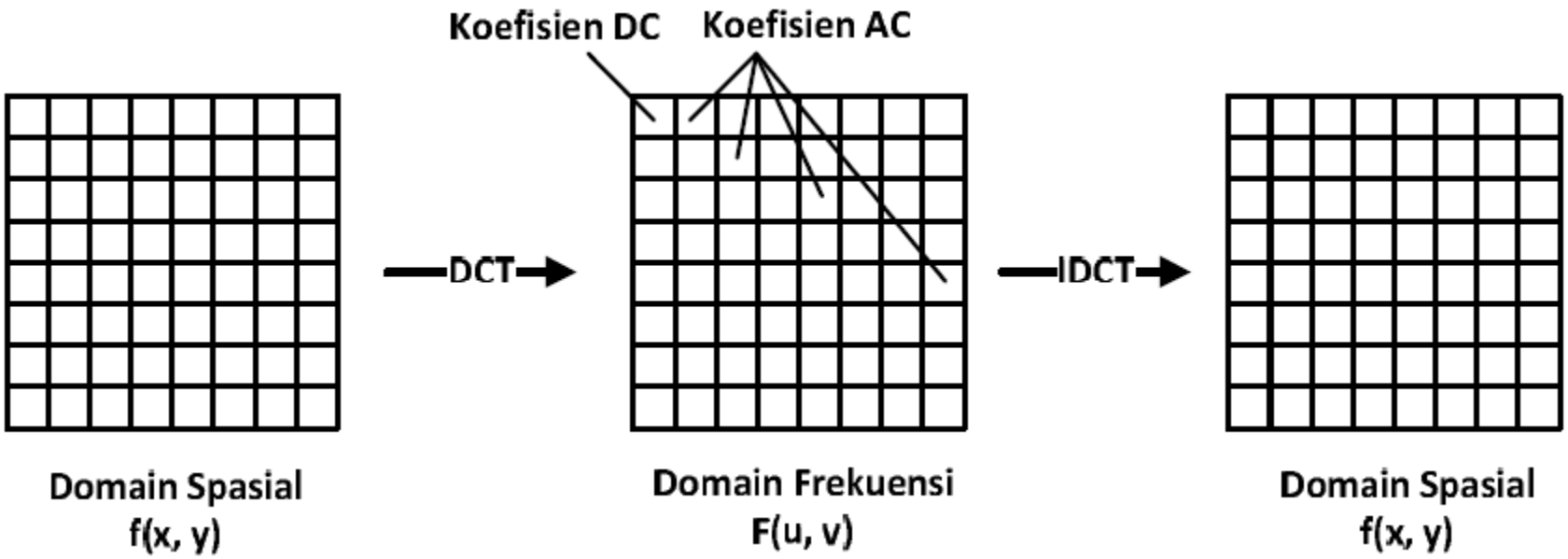
extracted watermark

Citra ber-*watermark* tidak dapat dibedakan dengan citra aslinya.

Bagaimana caranya?

- Tidak seperti metode *fragile watermarking* yang mana *watermark* disisipkan pada domain spasial (*pixel-pixel* citra),
- maka pada metode *robust watermarking*, *watermark* disisipkan pada doman frekuensi.
- Hal ini bertujuan agar *watermark* tahan terhadap manipulasi pada citra.

- Pertama-tama, citra ditransformasi dari ranah spasial ke ranah *transform* (frekuensi), misalnya menggunakan transformasi DCT (*Discrete Cosine Transform*)


Koefisien DC
Koefisien AC
DCT
IDCT
Domain Spasial
f(x, y)
Domain Frekuensi
F(u, v)
Domain Spasial
f(x, y)

- *Discrete Cosine Transform (DCT)*

$$C(u,v) = \alpha_u \alpha_v \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} I(x,y) \cos \frac{\pi(2x+1)u}{2M} \cos \frac{\pi(2y+1)v}{2N} \qquad (1)$$

$$\alpha_u = \begin{cases} \frac{1}{\sqrt{M}} & ,u = 0 \\ \sqrt{\frac{2}{M}} & ,1 \le u \le M-1 \end{cases} \qquad \alpha_v = \begin{cases} \frac{1}{\sqrt{N}} & ,v = 0 \\ \sqrt{\frac{2}{N}} & ,1 \le v \le N-1 \end{cases}$$

C(u,v) disebut koefisien-koefisien DCT

- *Inverse Discrete Cosine Transform (IDCT)*

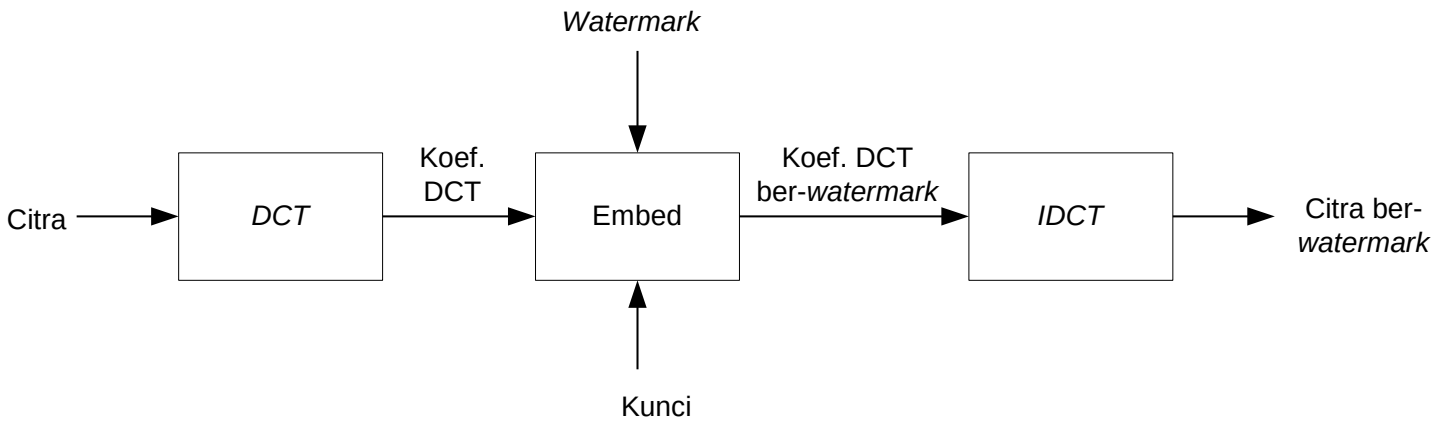
$$I(x,y) = \alpha_u \alpha_v \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} C(u,v) \cos \frac{\pi(2x+1)u}{2M} \cos \frac{\pi(2y+1)v}{2N} \qquad (4)$$

- Hasil tranformasi menghasilkan nilai-nilai yang disebut koefisien-koefisien transformasi (misalnya koefisien DCT).

- Bit-bit *watermark* (*w*) disembunyikan pada koefisien-koefisien tranformasi (*v*) tersebut dengan suatu formula:

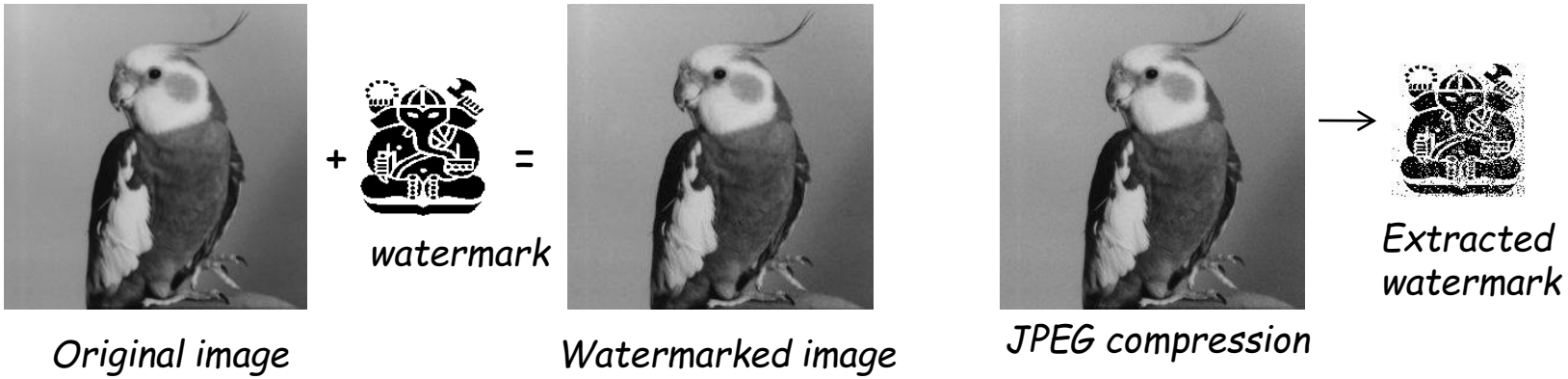
$$\hat{v}_i = v_i + w_i$$

- Selanjutnya, citra ditransformasikan kembali (*inverse transformation*) ke ranah spasial untuk mendapatkan citra stegano (atau citra *ber-watermark*).

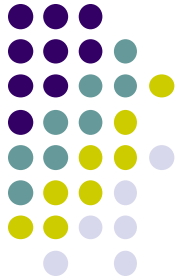


Test ketahahan *watermark* terhadap manipulasi terhadap citra.

Contoh: kompresi, *cropping*, *editing*, *resizing, dll*

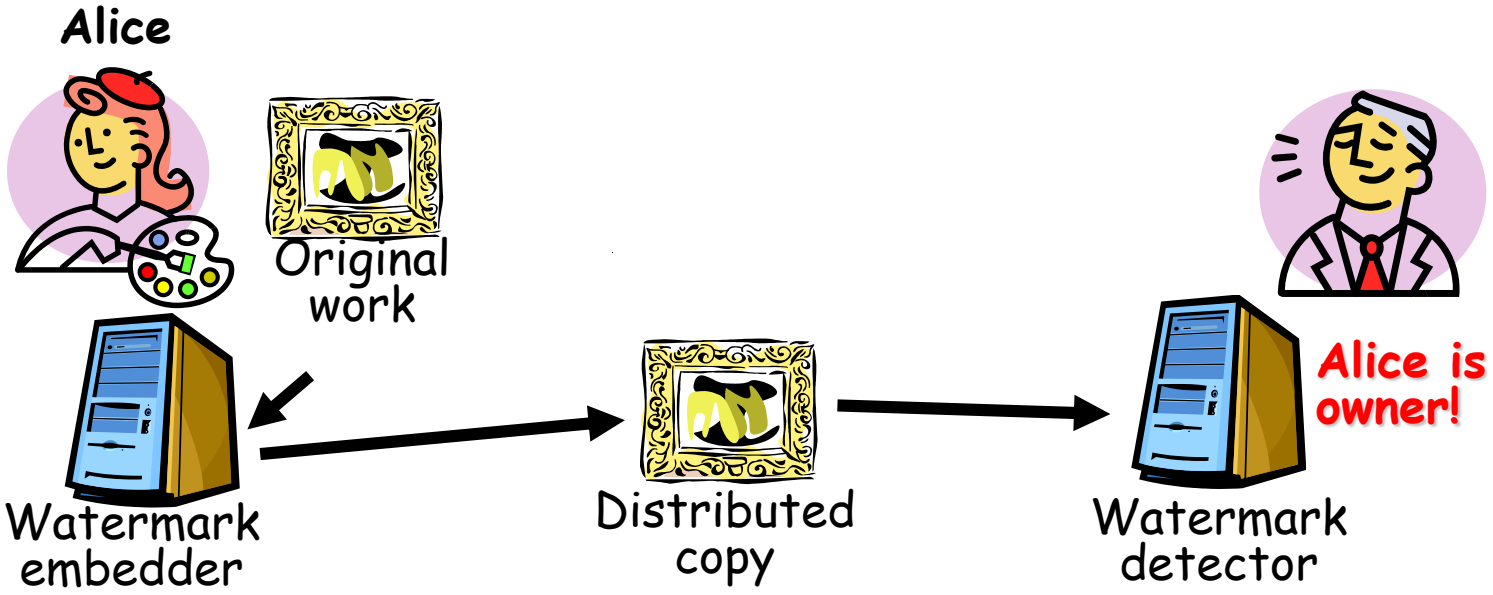


Aplikasi *Watermark*

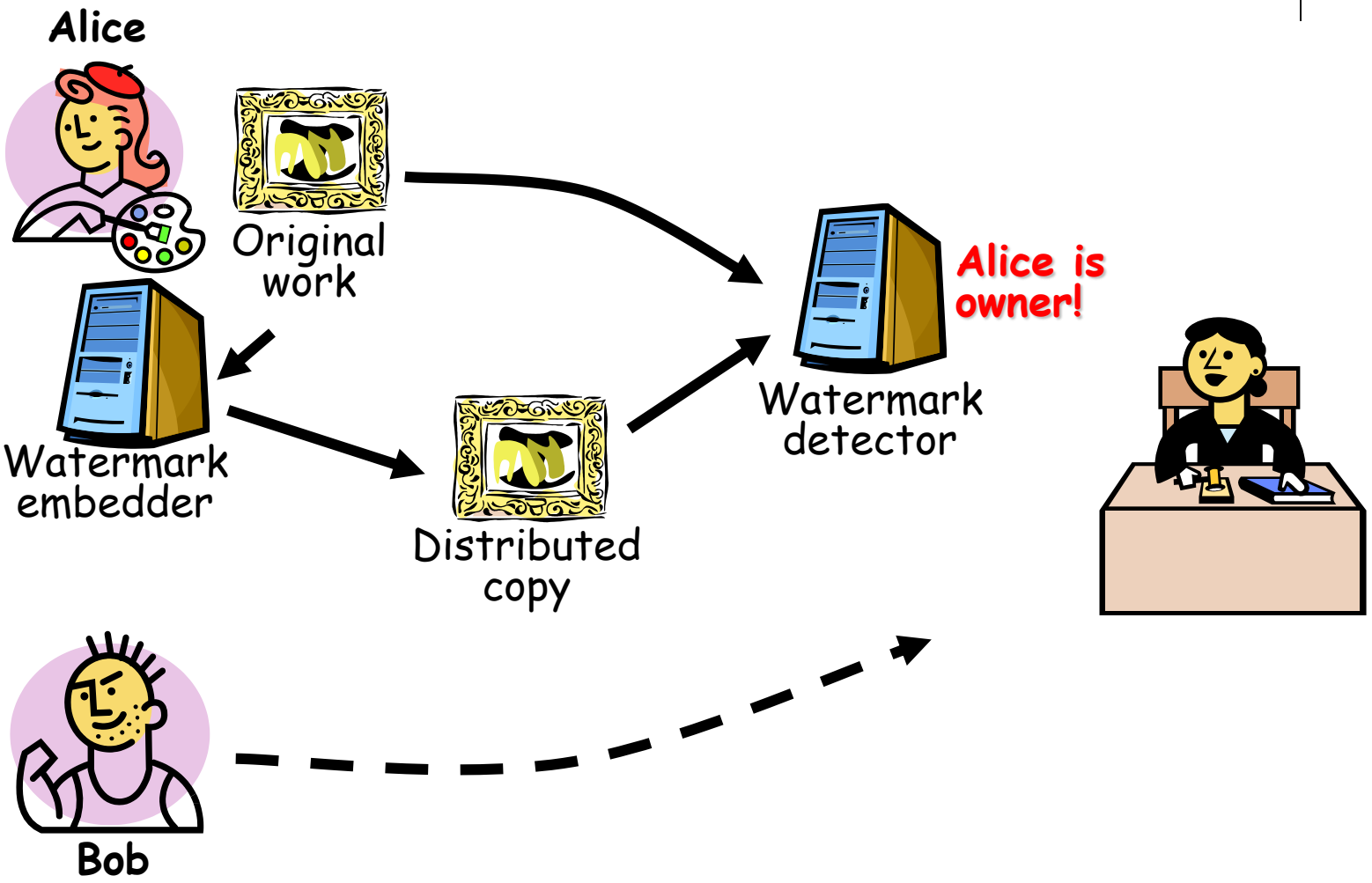


- Identifikasi kepemilikan (*ownership identification*)
- Bukti kepemilikan (*proof of ownership*)
- Memeriksa keaslian isi karya digital (*tamper proofing*) ➔ *Content authentication*
- *User authentication/fingerprinting/transaction tracking*: mengotentikasi pengguna spesifik. Contoh: distribusi DVD
- *Piracy protection/copy control*: mencegah penggandaan yang tidak berizin.
- Broadcast monitoring

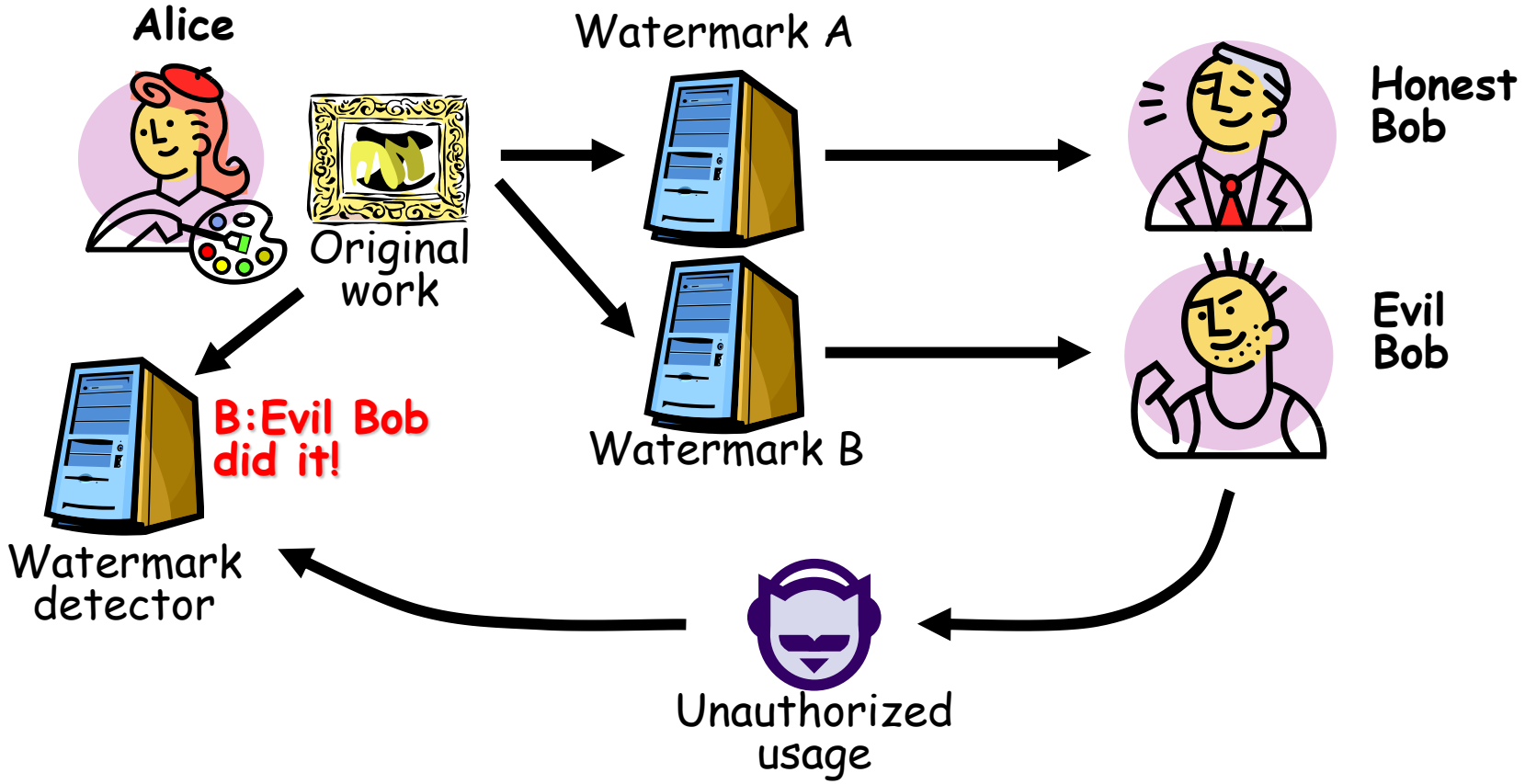
Aplikasi watermark: *Owner identification*



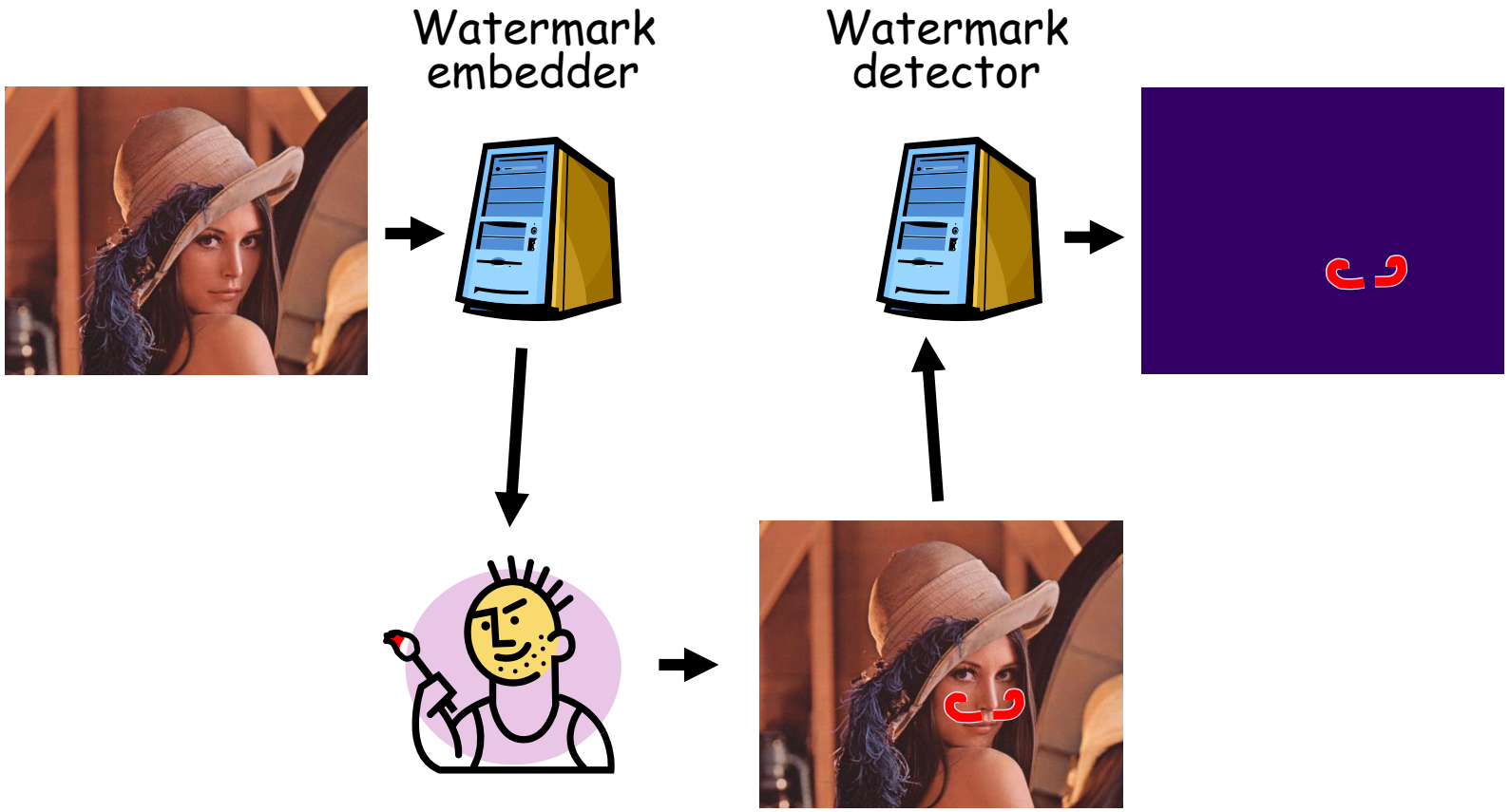
Aplikasi watermark: *Proof of ownership*



Aplikasi watermark: *Transaction tracking*

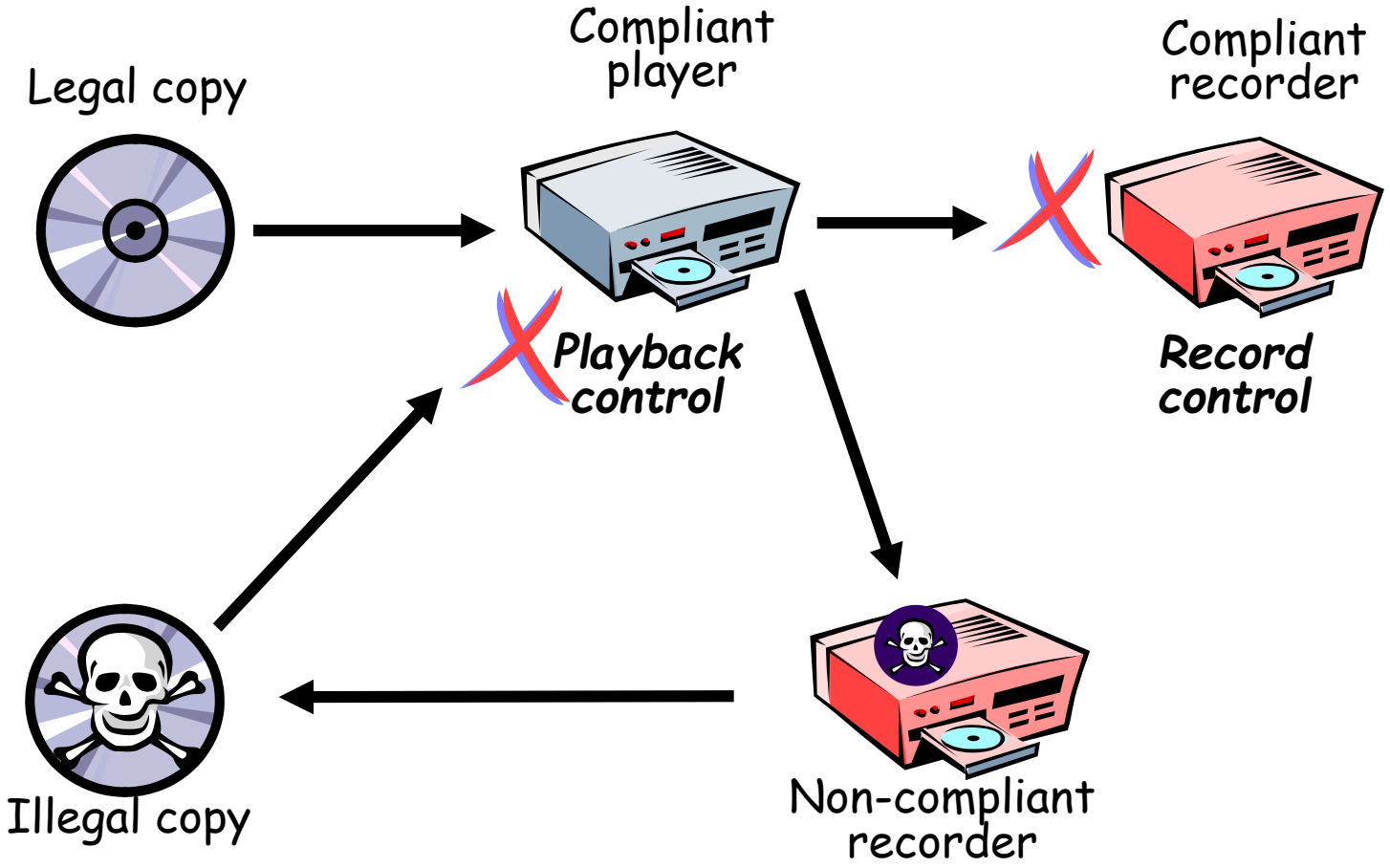


Aplikasi watermark: *Content authentication*



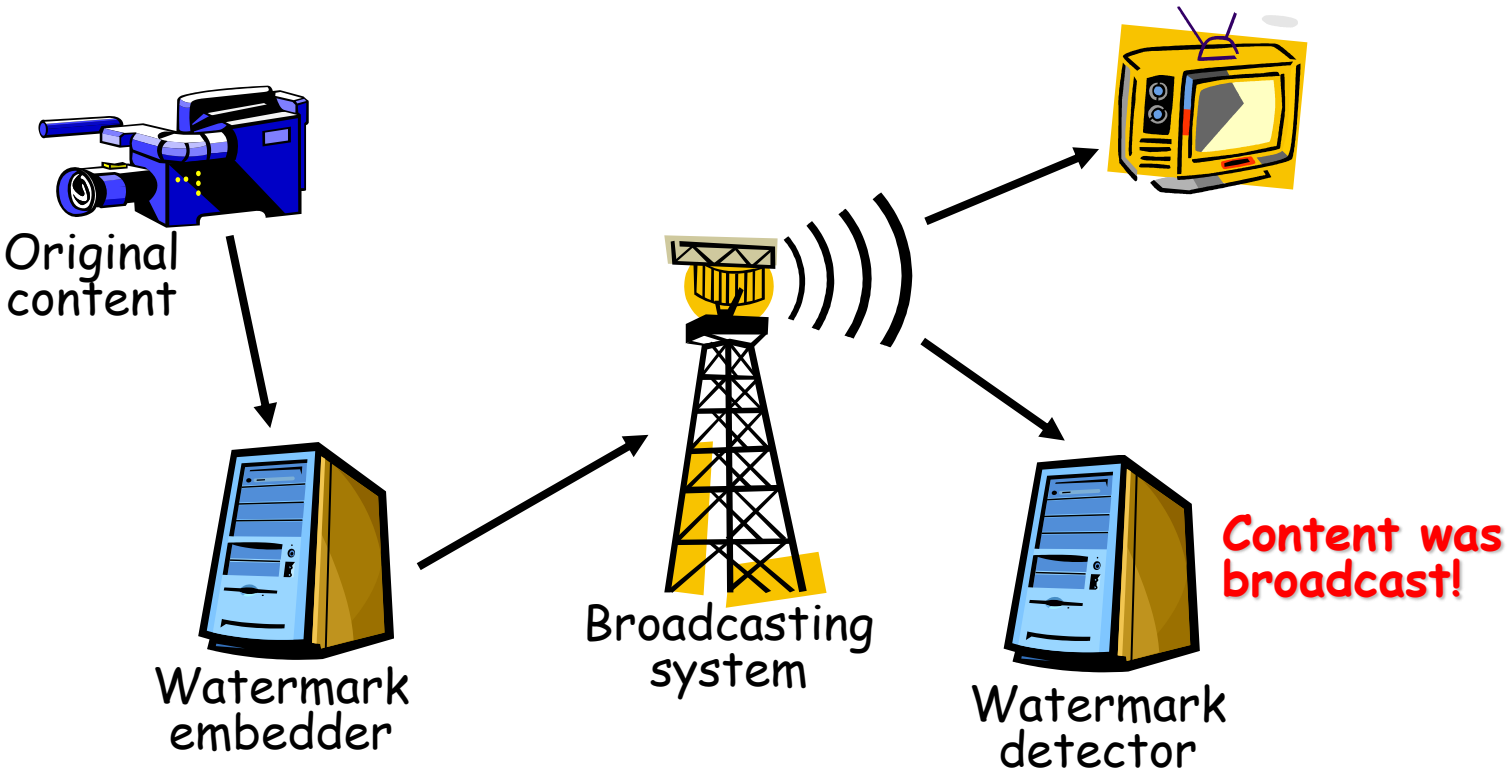
Aplikasi watermark: *Copy control/Piracy Control*

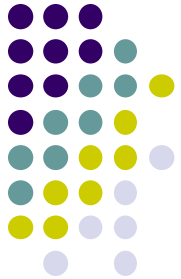
Watermark digunakan untuk mendeteksi apakah media digital dapat digandakan (copy) atau dimainkan oleh perangkat keras.



Aplikasi watermark: *Broadcast monitoring*

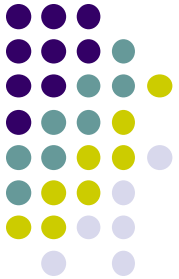
Watermark digunakan untuk memantau kapan konten digital ditransmisikan melalui saluran penyiaran seperti TV dan radio.





Tambahan

Metode *Spread Spectrum*



- *Watermark* disebar (*spread*) di dalam citra.

- *Spread spectrum* dapat dilakukan dalam 2 ranah:

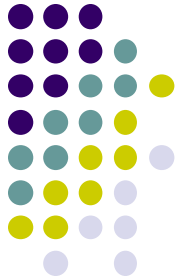
  1. Ranah spasial

     Menyisipkan *watermark* langsung pada nilai *byte* dari *pixel* citra.

  2. Ranah *transform*

     Menyisipkan *watermark* pada koefisien transformasi dari citra.

Spread Spectrum



- Merupakan bentuk komunikasi menggunakan frekuensi radio.
- Tujuannya untuk menyembunyikan informasi di dalam kanal frekuensi radio yang lebar sehingga informasi akan tampak seperti *noise*.
- Teknik *spread spectrum* mentransmisikan sinyal informasi pita-sempit (*narrow band signal information*) ke dalam sebuah kanal pita lebar dengan penyebaran frekuensi.
- Artinya, data yang dikirim dengan metode *spread spectrum* menyebar pada frekuensi yang lebar.

- Data yang disebar tampak terlihat seperti sinyal *noise* (*noise like signal*) sehingga sulit dideteksi, dicari, atau dimanipulasi.

- Metode *spread spectrum* awalnya digunakan di dalam militer, karena teknologi *spread spectrum* memiliki kemampuan istimewa sbb:

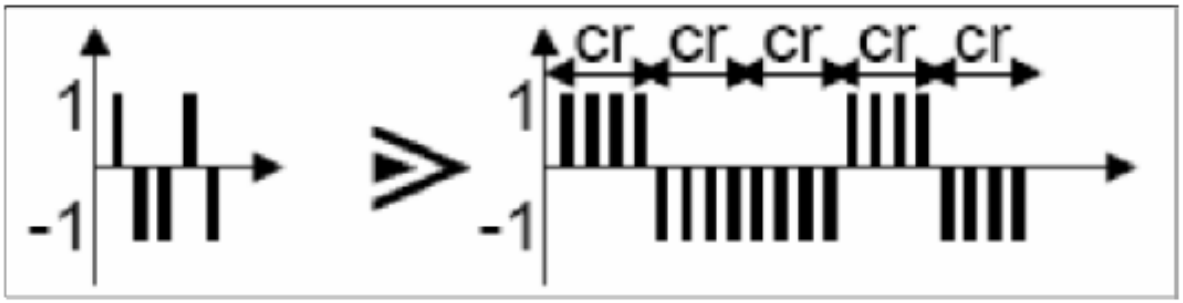
  1. Menyelundupkan informasi
  2. Mengacak data.

- Teknik *spread spectrum* ditemukan pada tahun 1930, dan dipatenkan pada tahun 1941 oleh Hedy Lamar dan George Antheil - *secret communications system used by the military.*

- Penyebaran data berguna untuk menambah tingkat redundansi.

- Besaran redundansi ditentukan oleh faktor pengali *cr* (singkatan dari *chip-rate*)

- Panjang bit-bit hasil redundansi menjadi *cr* kali panjang bit-bit awal.



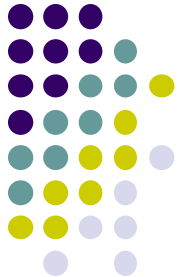
Before spreading *After spreading*

- Contoh: pesan = 10110 dan *cr* = 4, maka hasil *spreading* adalah 11110000111111110000

- Dengan teknik *spread spectrum*, data (pesan) dapat ditransmisikan tanpa penyembunyian tambahan, karena sudah tersembunyi dengan sendirinya.

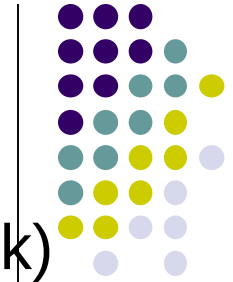
- Ide *spread spectrum* ini digunakan di dalam *watermarking* adalah untuk memberikan tambahan keamanan pada pesan dengan cara menempelkan pesan di dalam media lain seperti gambar, musik, video, atau artikel (teks).

Spread Spectrum Watermarking



- Pesan, yang disebut *watermark*, disisipkan ke dalam media dalam ranah frekuensi. Umumnya *watermark* berupa citra biner seperti logo atau penggalan musik.

- Penyisipan dalam ranah frekuensi membuat *watermark* lebih kokoh (*robust*) terhadap serangan (*signal processing*) ketimbang penyisipan dalam ranah spasial (seperti metode modifikasi LSB) → *robust watermarking*.



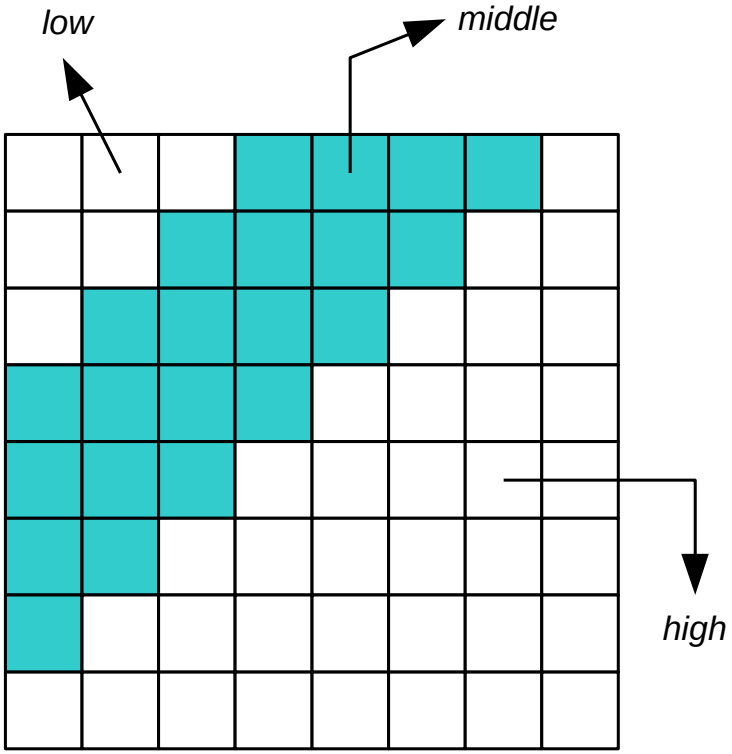
- Misalkan media yang akan disisipi pesan (watermark) adalah citra (image).

- Terlebih dahulu citra ditransformasi dari ranah spasial ke dalam ranah frekuensi.

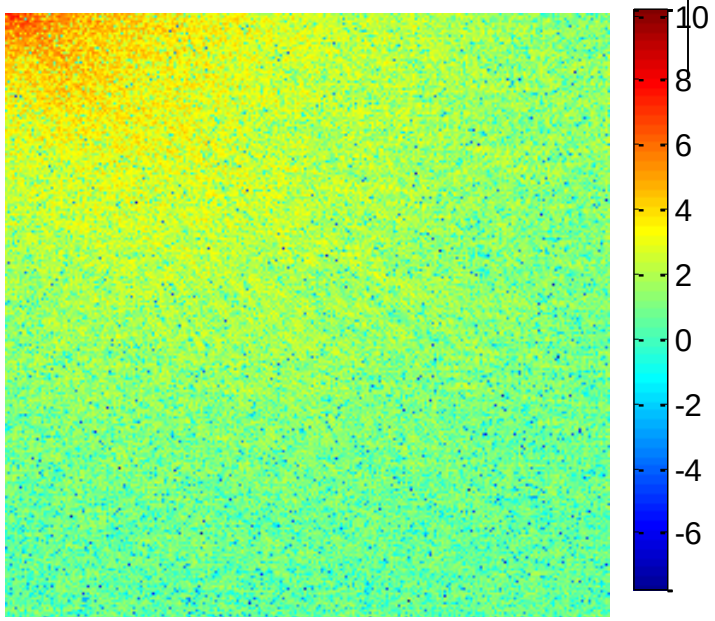
- Kakas transformasi yang digunakan antara lain:
  1. *Discrete Cosine Transform* (DCT)
  2. *Fast Fourier Transform* (FFT)
  3. *Discrete Wavelet Transform* (DWT)
  4. *Fourier-Mellin Transform* (FMT)

- Tinjau kakas transformasi yang digunakan adalah DCT.

- DCT membagi citra ke dalam 3 ranah frekuensi*: low frequencies*, *middle frequencies*, dan *high frequencies*)



Citra dalam ranah spasial



Citra dalam ranah frekuensi

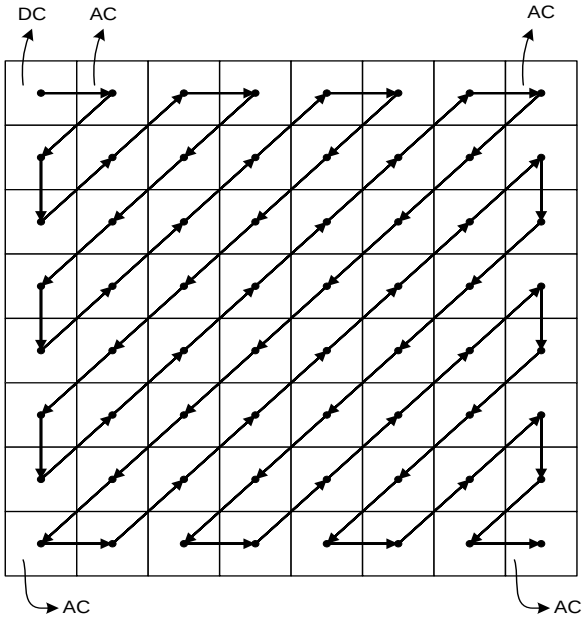
- Bagian *low frequency* berkaitan dengan tepi-tepi (*edge*) pada citra, sedangkan bagian *high frequency* berkaitan dengan detail pada citra.

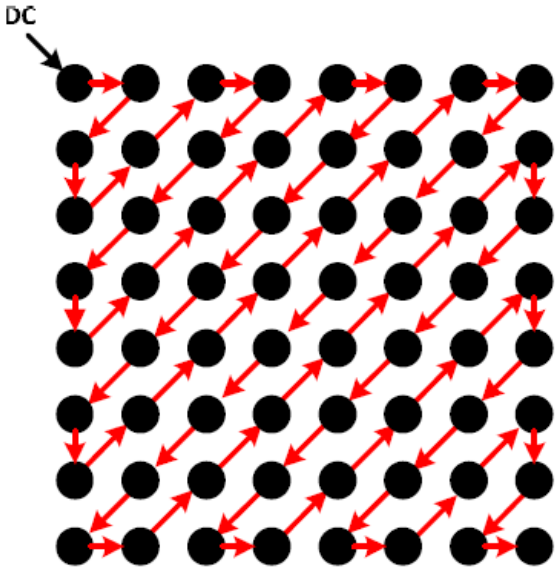
- Penyisipan pada bagian *low frequency* dapat merusak citra karena mata manusia lebih peka pada frekuensi yang lebih rendah daripada frekuensi lebih tinggi.

- Sebaliknya bila *watermark* disisipkan pada bagian *high frequency*, maka *watermark* tersebut dapat terhapus oleh operasi kuantisasi seperti pada kompresi *lossy* (misalnya *JPEG*).

- Oleh karena itu, untuk menyeimbangkan antara *robustness* dan *imperceptibility*, maka *watermark* disisipkan pada bagian *middle frequency* (bagian yang diarsir pada Gambar di atas).

- Bagian *middle frequency* diekstraksi dengan cara membaca matriks *DCT* secara *zig-zag* sebagaimana yang dilakukan di dalam algoritma kompresi *JPEG*





Pembacaan secara zigzag

Skema Penyisipan

1. Misalkan

$$A = \{a_i \mid a_i \in \{-1 , +1 \}\} \quad (1)$$

adalah bit-bit pesan (*watermark*) yang akan disembunyikan di dalam citra (catatan: bit 1 dinyatakan sebagai +1 dan bit 0 sebagai –1)

2. Setiap bit a_i dilakuan *spreading* dengan faktor *cr* yang besar, yang disebut *chip-rate*, untuk menghasilkan barisan:

$$B = \{b_i \mid b_i = a_j,\ j \cdot cr \le i < (j + 1) \cdot cr \}. \quad (2)$$

3. Bit-bit hasil *spreading* kemudian dimodulasi dengan barisan bit acak (*pseudo-noise):*

$$P = \{p_i \mid p_i \in \{-1 , 1 \}\} \quad (3)$$

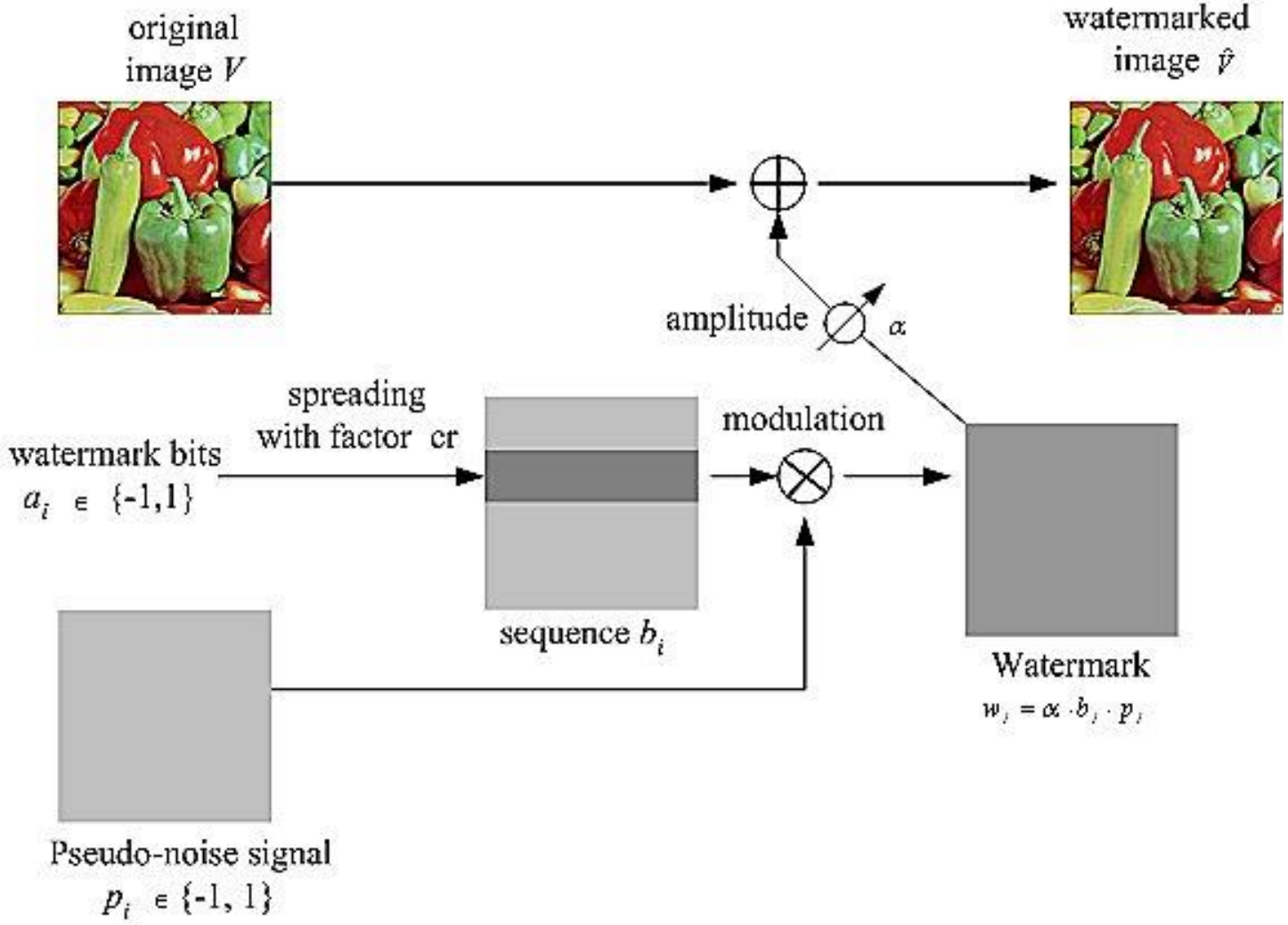
4. Bit-bit p_i diamplifikasi (diperkuat) dengan faktor kekuaran (strength) watermarking α untuk membentuk *spread spectrum watermark*

$$w_i = \alpha \cdot b_i \cdot p_i \qquad (4)$$

5. *Watermark* w_i disisipkan ke dalam citra (dalam ranah frekuensi) $V = \{v_i\}$ dengan persamaan:

$$\hat{v}_i = v_i + w_i \qquad (5)$$

Dikaitkan dengan sifat *noisy* p_i, w_i juga *a noise-like signal* dan sulit dideteksi, dicari, dan dimanipulasi.


original
image V
watermarked
image $\hat{v}$
amplitude
α
spreading
with factor cr
watermark bits
$a_i \in \{-1,1\}$
modulation
sequence b_i
Watermark
$w_j = \alpha \cdot b_j \cdot p_j$
Pseudo-noise signal
$p_i \in \{-1, 1\}$

Skema Ekstraksi

Untuk mengekstraksi pesan (*watermark*) dari citra stegano, penerima pesan harus memiliki *pseudo-noise pi* yang sama dengan yang digunakan pada waktu penyisipan.

Ekstraksi pesan dilakukan dengan cara sebagai berikut:

1. Kalikan citra stegano dengan p_i:

$$\sum_{i=j\cdot cr}^{(j+1)\cdot cr-1} p_i \cdot \hat{v}_i = \sum_{i=j\cdot cr}^{(j+1)\cdot cr-1} v_i \cdot p_i + \sum_{i=j\cdot cr}^{(j+1)\cdot cr-1} \alpha \cdot b_i \cdot p_i^2 \qquad (6)$$

Karena p_i acak, cr besar, dan deviasi v_i kecil, maka dapat diharapkan bahwa

$$\lim_{cr \to \infty} \sum_{i=j \cdot cr}^{(j+1) \cdot cr - 1} v_i \cdot p_i \approx 0 \qquad (7)$$

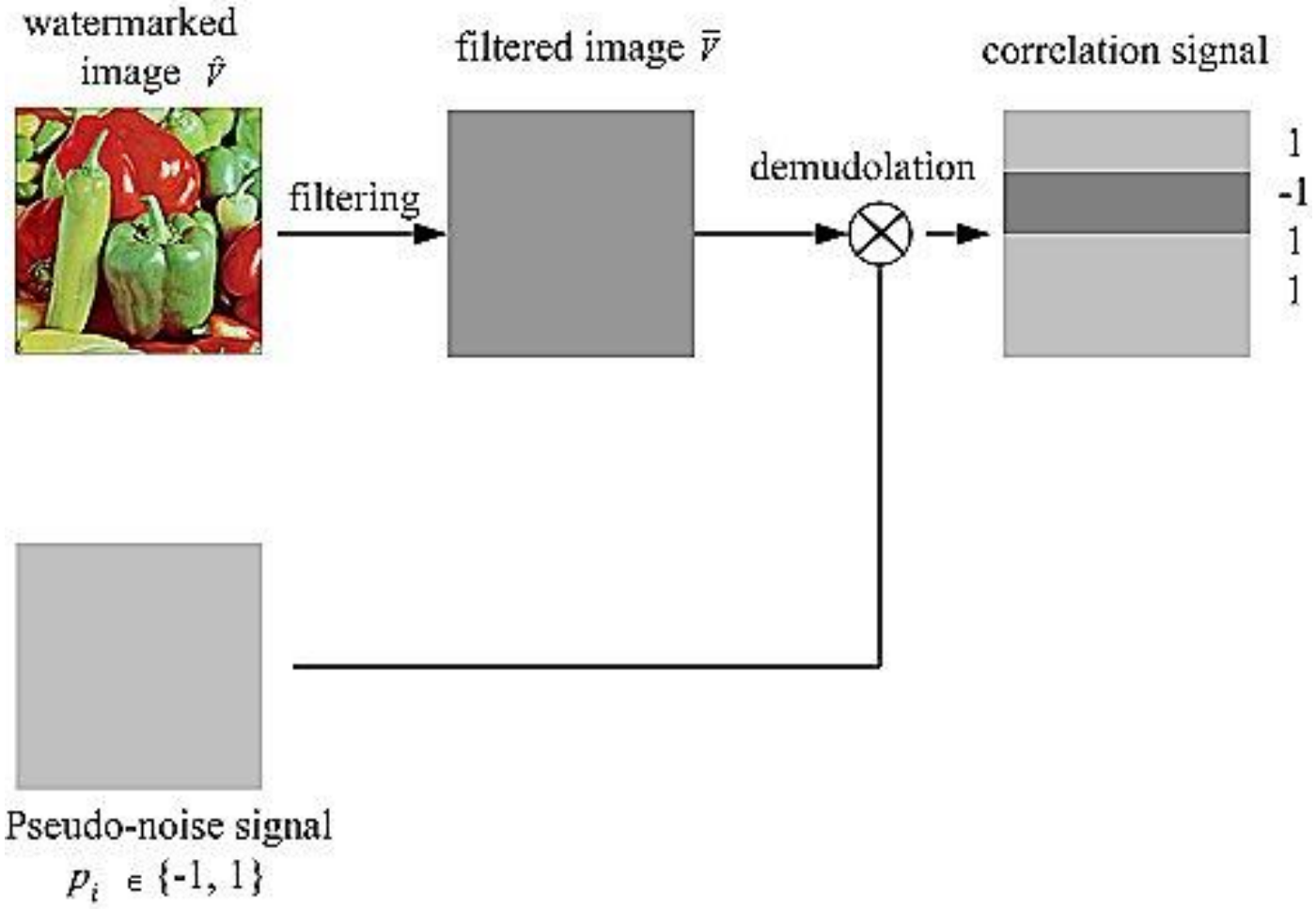
Karena $p_i^2 = 1$, persamaan (6) menghasilkan jumlah korelasi:

$$\sum_{i=j \cdot cr}^{(j+1) \cdot cr - 1} \hat{v}_i \cdot p_i = \alpha \cdot cr \cdot a_j \qquad (8)$$

Oleh karena itu, bit-bit yang disisipkan dapat ditemukan kembali dengan langkah 2 berikut:

2. Bit-bit pesan diperoleh kembali dengan persamaan berikut:

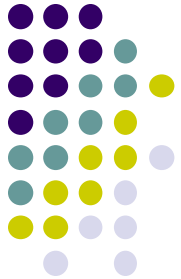
$$a_j = \begin{cases} 1, & \sum_{i=j\cdot cr}^{(j+1)\cdot cr-1} \hat{v}_i \cdot p_i > 0 \\ -1, & \sum_{i=j\cdot cr}^{(j+1)\cdot cr-1} \hat{v}_i \cdot p_i < 0 \end{cases} \qquad (9)$$


watermarked
image $\hat{v}$
filtered image $\bar{v}$
correlation signal
filtering
demudolation
1
-1
1
1
Pseudo-noise signal
$p_i \in \{-1, 1\}$

- Untuk membantu tahap korelasi, citra stegano dapat ditapis (*filtering*) terlebih dahulu dengan penapis lolos-tinggi seperti penapis Wiener atau penapis deteksi tepi.

- Penapisan dapat menghilangkan komponen yang timbul dari superposisi citra dan pesan (*watermark*).

Algoritma *Spread Spectrum Steganography*



A. Penyisipan pesan

1. Transformasi citra ke ranah frekuensi dengan menggunakan DCT. Simpan semua koefisien DCT di dalam matriks *M*.
2. Baca matrik *M* dengan algoritma *zigzag* untuk memperoleh koefisien-koefisien DCT, simpan di dalam vektor *V*.
3. *Spreading* pesan *A* dengan faktor *cr* untuk memperoleh barisan *B* dengan menggunakan persamaan (2). Misalkan panjang *B* adalah *m*.

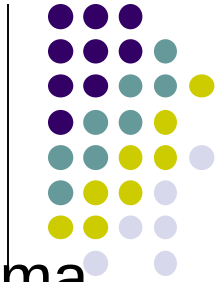
4. Bangkitkan barisan *pseudo-noise P* sepanjang *m*.
5. Kalikan p_i dan b_i dan α dengan persamaan (4) untuk menghasilkan w_i.

6. Sisipkan w_i ke dalam elemen-elemen *V* dengan persamaan (5). Untuk menyeimbangkan tingkat *imperceptibility* dan *robustness*, lakukan penyisipan pada *middle frequencies. Middle frequencies* dapat dipilih dengan melakukan lompatan pada *V* sejauh *L*.

7. Terakhir, terapkan IDCT untuk memperoleh citra stegano (*watermarked image*).

B. Ekstraksi pesan

1. Transformasi citra stegano ke ranah frekuensi dengan menggunakan DCT. Simpan semua koefisien DCT di dalam matriks *M*.
2. Baca matrik *M* dengan algoritma *zigzag* untuk memperoleh koefisien-koefisien DCT, simpan di dalam vektor *V*.
3. Bangkitkan barisan *pseudo-noise P* sepanjang *m*.
4. Kalikan p_i dan v_i dengan persamaan (6).
5. Dapatkan kembali bit-bit pesan (*watermark*) dengan persamaan (9).

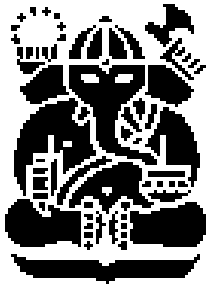


Catatan: Pesan yang diekstraksi tidak selalu tepat sama dengan pesan yang disisipkan, alasanya adalah:

1. DCT adalah transformasi yang *lossy*. Artinya, ada perubahan bit yang timbul selama proses transformasi. DCT beroperasi pada bilangan real. Operasi bilangan real tidak eksak karena mengandung pembulatan (*round-off*).
2. Bergantung pada awal posisi *middle frequency* yang digunakan (*L*). Posisi awal *middle frquency* hanya dapat diperkirakan dan tidak dapat ditentukan dengan pasti.

Original image

Stego-image



watermark

extracted watermark

Metode Cox

- Diusulkan pertama kali oleh Cox dalam makalah "*Secure Spread Spectrum Watermarking for Multimedia*" (1997).
- Watermark disisipkan pada komponen frekuensi (hasil transformasi DCT.
- Pada metode Cox, komponen frekuensi yang disisipi adalah komponen yang signifikan secara persepsi.
- Ada *trade-off* antara *robustness* dan *visibility* (α)

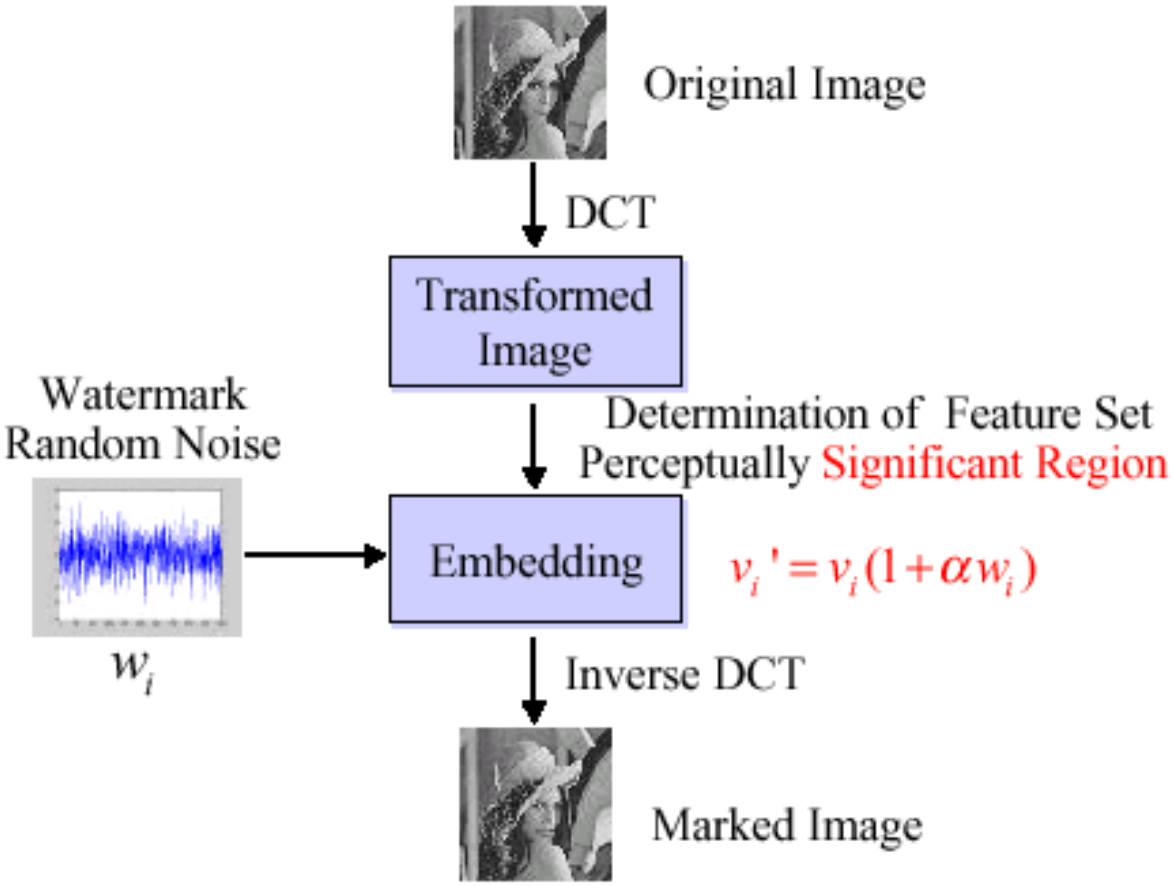
- *Watermark* $W = w_1, w_2, \ldots, w_n$
- *Watermark*: bilangan riil acak (*pseudo-noise*) yang mempunyai distribusi Normal:

$$p(w) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{w^2}{2\sigma^2}\right)$$

- Cox memilih *watermark* mempuyai distribusi $N(0, 1)$, yaitu *mean* = 0, variansi = 1.

- Menurut Cox, *watermark* tsb mempunyai kinerja lebih baik daripada data yang terdistribusi *uniform*.

- Penyisipan *watermark*:



- Pendeteksian *watermark*:

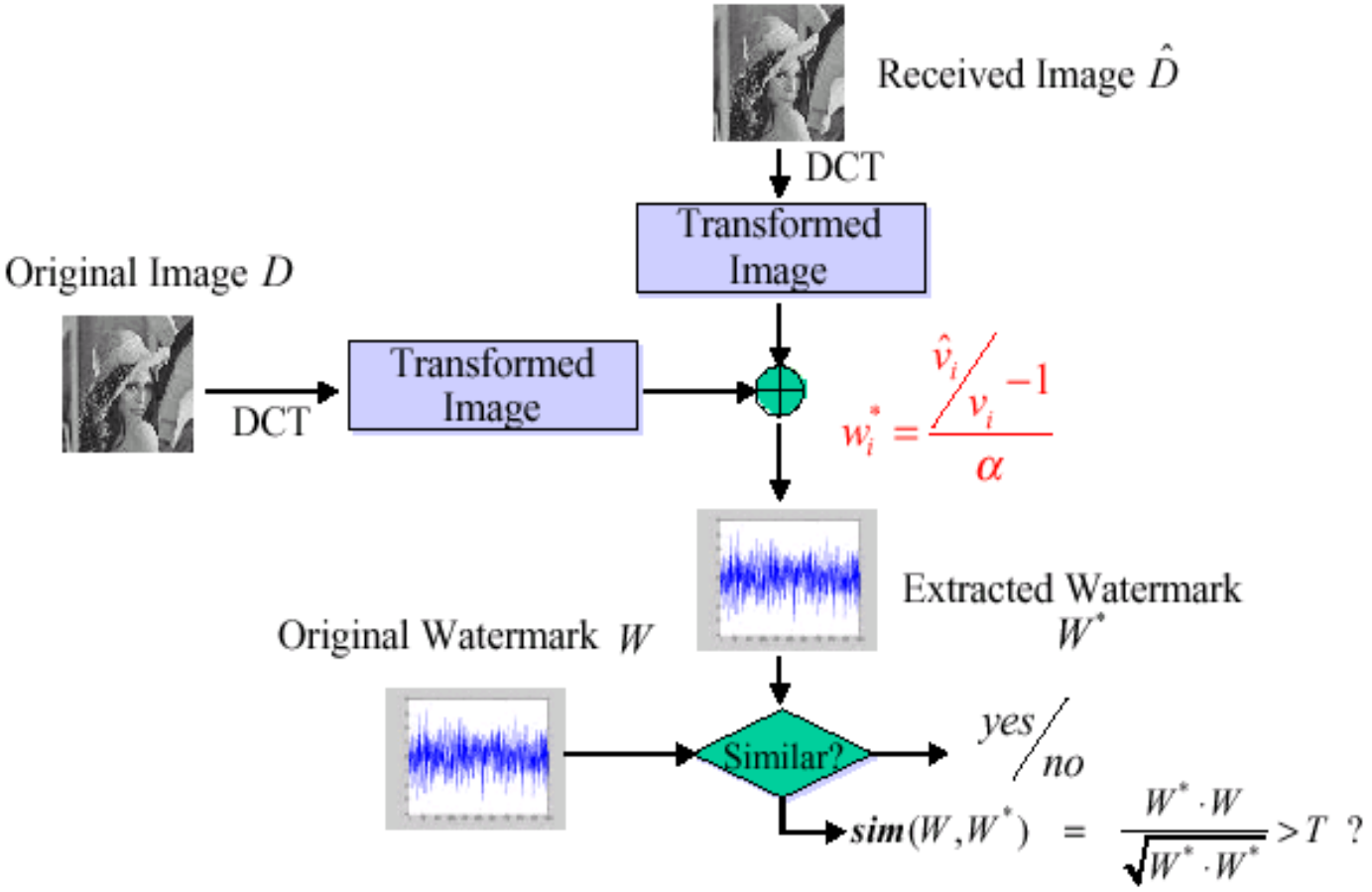
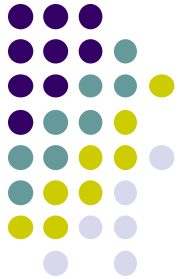


Fig. 4. Bavarian couple image courtesy of Corel Stock Photo Library.

Fig. 5. Watermarked version of Bavarian couple.



- Panjang *watermark* = *n* = 1000
- Cox menggunakan 1000 koefisien terbesar. Inilah yang dinamakan *frequency spreading*.
- Cox memilih α = 0.1 dan *T* = 6
- Kelemahan: perlu citra asli untuk deteksi *watermark (non-blind watermarking)*.
- Kelebihan: kokoh terhadap
  - konversi analog-ke-digital
  - Konversi digital-ke-analog
  - *Cropping*
  - Kompresi, rotasi, translasi, dan penskalaan

Referensi

1. Nick Sterling, Sarah Wahl, Sarah Summers, *Spread Spectrum Steganography*.
2. F. Hartung, and B. Girod, *Fast Public-Key Watermarking of Compressed Video*, Proceedings of the 1997 International Conference on Image Processing (ICIP ’97).
3. Winda Winanti, *Penyembunyian Pesan pada Citra Terkompresi JPEG Menggunakan Metode Spread Spectrum*, Tugas Akhir Informatika ITB, 2009.