Proteksi Citra Foto KPM Mahasiswa Fasilkom UNSRI dari *DeepFake* dengan CMUA-*Watermark*

Diajukan untuk Menyusun Skripsi
di jurusan Teknik Informatika Fakultas Ilmu Komputer UNSRI

Oleh :

Renaldi Budi Setiawan
NIM : 09021281823066

Jurusan Teknik Informatika

FAKULTAS ILMU KOMPUTER UNIVERSITAS SRIWIJAYA

2023

LEMBAR PENGESAHAN PROPOSAL SKRIPSI

Proteksi Citra Foto KPM Mahasiswa Fasilkom UNSRI dari *DeepFake* dengan CMUA-*Watermark*

Oleh :

Renaldi Budi Setiawan

NIM : 09021281823066

Indralaya, 26 mei 2023

Pembimbing I	Pembimbing II,
Syamsuryadi, S.Si., M.Kom., Ph.D.	Muhammad Qurhanul Rizqie, S.KOM,. M.T., Ph.D.
NIP 197102041997021003	NIP 1671060312870008

Mengetahui,
Ketua Jurusan

Alvi Syahrini Utami, M.Kom.
NIP. 19781222200642003

DAFTAR ISI

Halaman

DAFTAR TABEL

Halaman

DAFTAR GAMBAR

Halaman

BAB I

PENDAHULUAN

1.1 Pendahuluan

Pada bab ini akan dibahas berkenaan dengan garis besar pokok-pokok pikiran dalam penelitian ini. Pokok pikiran yang akan dibahas antara lain latar belakang masalah, rumusan masalah, batasan masalah, tujuan penelitian, dan manfaat penelitian. Pokok-pokok pikiran yang diuraikan akan dijadikan acuan dalam kajian penelitian ini.

1.2 Latar Belakang Masalah

Baru-baru ini, peningkatan *Generative Adversarial Networks* (GAN) telah menunjukkan hasil yang mengesankan dalam pembuatan konten virtual, menciptakan nilai ekonomi dan hiburan yang cukup besar. Namun, *deepfake*s, jaringan modifikasi wajah berbasis pembelajaran mendalam yang menggunakan GAN untuk menghasilkan konten palsu dari orang yang ditargetkan atau atribut target, telah menyebabkan kerusakan besar pada privasi dan reputasi orang. Di satu sisi, gambar dan video palsu dapat menunjukkan hal-hal yang tidak pernah dikatakan atau dilakukan oleh seseorang, sehingga merusak reputasinya, terutama jika melibatkan pornografi atau politik (Tolosana et al., 2020). Di sisi lain, gambar wajah palsu dengan atribut target dapat melewati otentikasi biometrik aplikasi komersial, yang berpotensi melanggar keamanan (Korshunov & Marcel, 2018). Oleh karena itu, mempertahankan ancaman yang dibawa oleh *deepfake*s tidak hanya membutuhkan distorsi gambar yang dimodifikasi dan menurunkan kualitas

visualnya untuk membantu manusia dalam membedakannya dari gambar yang realistis, tetapi juga memastikan bahwa wajah palsu tidak lolos deteksi kehidupan, yang merupakan langkah pertama dari sebagian besar verifikasi biometrik.

Situs resmi milik UNSRI versi lama[1] hingga saat ini dapat dengan mudah diakses oleh siapapun tanpa memerlukan verifikasi terlebih dahulu. Adanya laman ini juga memberikan informasi terkait mahasiswa, termasuk didalamnya foto diri mahasiswa. Setiap foto yang diunggah oleh pihak UNSRI di situs tersebut akan muncul dalam hasil pencarian gambar Google dan dapat dengan mudah diunduh. Informasi pribadi mahasiswa yang ada di dalam situs ini sangat memungkinkan dapat dieksploitasi oleh seseorang untuk melakukan tindakan kejahatan, seperti mengatasnamakan identitas dan menggunakan wajah mahasiswa tersebut dengan menggunakan *deepfake*.

Adversarial watermark dapat digunakan untuk memerangi *deepfake model*, *adversarial watermark* dapat menghasilkan citra gambar yang terdistorsi (Ruiz et al., 2020). Namun metode ini masih kurang efisien karena memerlukan proses pelatihan individu untuk setiap citra gambar wajah, untuk menghasilkan *adversarial attack model* terhadap *deepfake model* tertentu (Huang et al., 2021). Untuk mengatasi masalah ini, penelitian ini menggunakan metode *universal adversarial attack model* pada *deepfake model*, untuk menghasilkan *Cross-Model Universal Adversarial Watermark* (CMUA-*Watermark*) yang dapat melindungi ribuan citra gambar wajah dari beberapa model *deepfake* (Huang et al., 2021).

[1] https://old.UNSRI.ac.id/?act=daftar_mahasiswa

1.3 Rumusan Masalah

Berdasarkan permasalahan pada latar belakang yang telah diuraikan maka rumusan masalah dari penelitian ini adalah

1. Membuat kerangka kerja penelitian proteksi citra gambar foto KPM mahasiswa Fasilkom UNSRI dari *deepfake* dengan CMUA-*Watermark.*
2. Bagaimana cara membangun perangkat lunak yang dapat memproteksi citra gambar foto KPM mahasiswa Fasilkom UNSRI dari *deepfake* menggunakan metode CMUA-*Watermark.*
3. Bagaimana tingkat keberhasilan metode CMUA-*Watermark* dalam memproteksi citra foto KPM mahasiswa Fasilkom UNSRI dari *deepfakes*?

1.4 Tujuan Penelitian

Tujuan penelitian ini adalah:

1. Menghasilkan kerangka kerja yang sesuai untuk penelitian proteksi citra gambar foto KPM mahasiswa Fasilkom UNSRI dari *deepfake* dengan CMUA-*Watermark.*
2. Menghasilkan perangkat lunak yang dapat memproteksi citra gambar foto KPM mahasiswa Fasilkom UNSRI dari *deepfake* menggunakan metode CMUA-*Watermark.*
3. Mengetahui tingkat keberhasilan penggunaan metode CMUA-*Watermark* dalam memproteksi citra foto KPM mahasiswa Fasilkom UNSRI dari *deepfake*.

1.5 Manfaat Penelitian

Manfaat penelitian ini adalah:

1. Sistem yang dibuat dapat memproteksi citra gambar foto KPM mahasiswa Fasilkom UNSRI dari *deepfake* menggunakan metode CMUA-*Watermark.*
2. Hasil penelitian dapat dijadikan sebagai rujukan untuk penelitian terkait di masa mendatang.

1.6 Batasan Masalah

Batasan masalah dalam penelitian ini adalah sebagai berikut:

1. *Dataset* yang digunakan pelatihan merupakan *dataset* Celeb-a, dari penelitian *Deep Learning Face Attributes in the Wild* (2015).
2. Data uji yang digunakan merupakan *dataset* foto mahasiswa Fakultas Ilmu Komputer Universitas Sriwijaya Angkatan 2018.
3. Ekstensi citra yang didukung oleh perangkat lunak adalah .jpg.
4. Penelitian hanya fokus dalam proteksi citra dari *deepfake*.

1.7 Sistematika Penulisan

Sistematika penulisan tugas akhir mengikuti standar penulisan tugas akhir Fakultas Ilmu Komputer Universitas Sriwijaya yaitu sebagai berikut:

BAB I. PENDAHULUAN

Pada bab ini akan membahas landasan dari penelitian, seperti latar belakang, rumusan masalah, tujuan dan manfaat penelitian, batasan masalah serta sistematika penulisan.

BAB II. KAJIAN LITERATUR

Pada bab ini membahas literatur pada penelitian, seperti pengertian Citra, *Deepfake,* CMUA-*Watermark* dan penelitian yang relevan.

BAB III. METODOLOGI PENELITIAN

Pada Bab ini menjelaskan pelaksanaan alur penelitian, yakni pengumpulan data dan perancangan pembangunan perangkat lunak. Serta tahapan dijelaskan secara detail berdasarkan kerangka yang dibuat.

1.8 Kesimpulan

Pada Bab ini telah menjelaskan dasar dan patokan pada penelitian , seperti latar belakang, rumusan masalah, tujuan penelitian, manfaat penelitian, batasan masalah dan sistematika penulisan.

BAB II

KAJIAN LITERATUR

2.1 Pendahuluan

pada bab ini akan dijelaskan mengenai dasar-dasar teori digunakan pada penelitian ini. Serta penjelasan hasil dari penelitian-penelitian terkait mengenai citra, *Deepfakes*, CMUA-*Watermark* dan RUP. Pada bab ini pula dibahas mengenai penelitian terkait lainnya yang relevan.

2.2 Landasan Teori

2.2.1 Citra

Citra merupakan representasi visual dari objek atau *scene* yang dibentuk oleh kumpulan piksel-piksel (Rafique et al., 2018). Citra digital terdiri dari matriks piksel, di mana setiap piksel mewakili intensitas cahaya atau warna pada posisi tertentu dalam citra (Gonzalez & Woods, 2018). Citra dapat berupa citra *grayscale* yang hanya memiliki tingkat keabuan (*grayscale*) atau citra berwarna yang terdiri dari tiga komponen warna dasar (merah, hijau, biru) yang membentuk citra dalam model warna RGB (*Red-Green-Blue*) (M. Sonka, 2014).

2.2.1.1 Peningkatan dan Restorasi Citra

Peningkatan citra adalah proses memperbaiki atau meningkatkan kualitas citra dengan memperjelas detail, meningkatkan kontras, atau mengurangi *noise* (M. Sonka, 2014). Restorasi citra adalah proses pemulihan citra yang terdegradasi akibat gangguan atau kerusakan, seperti *blur* atau *noise* (Gonzalez & Woods,

2018). Teknik umum yang digunakan dalam peningkatan dan restorasi citra termasuk filter spasial, filter frekuensi, atau metode restorasi berbasis statistik.

2.2.1.2 *Image Watermarking*

Image watermarking adalah teknik yang digunakan untuk menambahkan informasi rahasia ke citra gambar digital yang tidak dapat dikenali secara visual (Rafique et al., 2018). Tujuan dari watermark gambar adalah untuk memberikan integritas, keaslian dan keotentikan pada citra agar dapat melindungi hak cipta dan mencegah pemalsuan (Gonzalez & Woods, 2018).

Adapun *watermarking* Berbasis Perceptual Teknik ini mempertimbangkan karakteristik perseptual manusia dalam menyisipkan watermark, dengan memperhatikan batas toleransi manusia terhadap perubahan visual yang terlihat (Cox & Miller, 2001). Teknik ini memungkinkan penyisipan yang lebih tidak terlihat oleh mata manusia.

2.2.2 *DeepFake*

Deepfake sendiri baru dipopulerkan di tahun 2017, berawal dari pengguna *Reddit* mengunggah video porno hasil editan. Pengguna *Reddit* ini mengembangkan GAN menggunakan *TensorFlow*. Teknologi *Deepfake* dapat berupa video lucu, pornografi, atau politik seseorang yang mengatakan apa pun, tanpa persetujuan orang yang citra gambar dan suaranya terlibat (Day, 2019; Fletcher, 2018).

2.2.2.1 *Photo Deepfake*

2.2.2.1.1 *Face and Body Swapping*

Dalam hal ini, perubahan dilakukan pada wajah dan tubuh dengan mengganti atau memadukan tubuh dan wajah dengan wajah atau tubuh orang lain. Hasilnya adalah orang yang sama sekali berbeda dalam citra gambar aslinya. Contoh pendekatan ini dapat dilihat di banyak aplikasi menggunakan *Aging filter*. Ini dapat berguna bagi pelanggan untuk mencoba pakaian, kosmetik, atau gaya rambut secara virtual.

2.2.2.2 *Photo Deepfake Creation*

Pada tahap pembuatan *Photo Deepfake*, langkah-langkah umum yang dilakukan adalah sebagai berikut:

1. Ekstraksi Wajah dan Tubuh: Dalam tahap ini, wajah dan tubuh dari citra asli diekstraksi dengan menggunakan metode deteksi wajah dan tubuh yang tepat (Rafique et al., 2018).
2. Pemrosesan dan Transformasi: Wajah dan tubuh yang diekstraksi kemudian diproses dan ditransformasi untuk mengubah fitur dan proporsi agar sesuai dengan wajah atau tubuh orang lain yang akan digunakan (Rafique et al., 2018). Teknik seperti transformasi geometri dan pemetaan fitur wajah digunakan untuk mencapai hasil yang akurat.
3. Penggabungan dan Penyesuaian: Setelah wajah dan tubuh yang dimodifikasi siap, mereka digabungkan kembali dengan citra asli dengan menggantikan atau memadukan wajah dan tubuh asli (Rafique et al., 2018). Tahap ini

memerlukan teknik *blending* dan penyesuaian warna untuk mencapai kesan yang realistis.

4. Evaluasi dan Kualitas: Tahap ini melibatkan evaluasi visual terhadap hasil *Photo Deepfake* yang dibuat untuk memastikan keaslian dan kualitas yang memadai (Rafique et al., 2018).

2.2.3 CMUA-*Watermark*

CMUA-*Watermark* adalah sebuah teknik *watermarking* yang digunakan untuk memberikan tanda air (*watermark*) pada data multi-media secara rahasia dan tangguh. Metode ini juga memiliki kemampuan untuk mengatasi masalah kehilangan atau perubahan data yang disebabkan oleh proses kompresi, *cropping*, rotasi, dll (Li et al., 2019).

2.2.3.1 Metode CMUA-*Watermark*

Adapun beberapa metode yang digunakan dalam CMUA-*Watermark:*

2.2.3.1.1 *Universal Adversarial Perturbation*

Universal Adversarial Perturbation (UAP) adalah sebuah jenis perturbation yang dihasilkan dengan tujuan mengelabui model machine learning secara universal. UAP adalah perubahan yang ditambahkan pada data input yang mampu mempengaruhi model secara konsisten dan dapat mengecoh model untuk menghasilkan prediksi yang salah (Moosavi-Dezfooli et al., 2017).

UAP biasanya dihasilkan dengan menggunakan algoritma optimisasi seperti teknik *gradient descent* atau teknik evolusi. *Perturbation* ini bersifat universal karena dapat diterapkan pada berbagai data *input* tanpa memerlukan

informasi khusus tentang data tersebut. UAP cenderung menjadi noise yang tidak terlihat oleh manusia, sehingga sulit untuk mendeteksi keberadaannya.

2.2.3.1.2 *Mean Square Error* (MSE)

Mean Square Error (MSE) adalah metrik yang umum digunakan untuk mengukur kesalahan atau perbedaan antara dua set data atau citra. Dalam konteks *watermarking*, MSE dapat digunakan untuk mengukur sejauh mana perubahan watermark pada citra yang ditandai dari citra asli. MSE dihitung dengan mengambil selisih kuadrat dari setiap piksel pada citra yang ditandai dan citra asli, kemudian menghitung rata-rata dari selisih kuadrat tersebut (Gonzalez & Woods, 2018). Semakin kecil nilai MSE, semakin sedikit perubahan atau distorsi yang terjadi pada citra yang ditandai. Metode MSE disini digunakan untuk mengukur perbedaan antara $G(I_1)...G(I_n)$(gambar yang *deepfake* tanpa proteksi) dan $G(I_1 + W)…G(I_n + W)$(gambar yang telah diproteksi terlebih dahulu sebelum di deepfake), di mana *E* adalah nilai batas atas dari *adversarial watermark W*.

$$\max_{W} \sum_{i=1}^{n} MSE(G(I_i), G(I_i + +W)), s.t. ||W||_\infty \leq \epsilon,$$

2.2.3.1.3 PGD (*Projected Gradient Descent*)

PGD adalah algoritma optimisasi yang digunakan untuk menghasilkan *adversarial perturbation* pada data, terutama pada konteks serangan terhadap model *deep learning*. *Adversarial perturbation* adalah perubahan yang disisipkan pada data dengan tujuan mengelabui model dan mempengaruhi hasil prediksi yang salah (Madry et al., 2018). PGD bekerja dengan menghitung gradien fungsi biaya terhadap data input dan mengubah nilai data input dalam arah gradien untuk

menghasilkan *perturbation*. Dalam CMUA-watermark metode ini digunakan sebagai *attack base* untuk memperbarui *adversarial perturbation* pada setiap iterasi *attack*, di mana *I* adalah citra wajah yang bersih, I^r_{adv} adalah *adversarial facial images* pada iterasi ke-*r*, *a* adalah *step size* dari *base attack*, *L* adalah *loss function*, *G* adalah *face modification network* yang di *attack*, dan *clip* operasi membatasi I_{adv} pada rentang $[I - \epsilon, I + \epsilon]$.

$$I^0_{adv} = I + W,$$

$$I^{r+1}_{adv} = clip_{I,\epsilon}\{I^r_{adv} + a\ sign(\nabla_I L(G(I^r_{adv}), (G(I)))\}.$$

2.2.3.1.4 *Adversarial Perturbation Fusion*

Adversarial Perturbation Fusion adalah teknik yang digunakan untuk menggabungkan beberapa *perturbation* yang dihasilkan oleh metode *adversarial learning* atau serangan terhadap data. Teknik ini bertujuan untuk meningkatkan efektivitas serangan atau dampak perubahan yang dihasilkan pada citra target (Jain et al., 2020). Dalam konteks *watermarking*, Adversarial Perturbation Fusion dapat digunakan untuk meningkatkan ketangguhan watermark terhadap serangan atau upaya penghapusan. Konflik di antara watermark yang berlawanan yang dihasilkan dari gambar dan model yang berbeda akan mengurangi kemampuan transferabilitas CMUA-*Watermark* yang diusulkan. Untuk melemahkan konflik ini, digunakan strategi fusi gangguan dua tingkat selama proses serangan. Secara khusus, ketika menyerang satu model *deepfake* tertentu, akan melakukan **fusi tingkat gambar** untuk merata-rata gradien yang di *sign* dari sekumpulan gambar wajah,

$$G_{avg} = \frac{\sum_j^{bs} sign(\nabla_{I_j} L\left(G\left(I_j^{adv}\right), G\left(I_j\right)\right))}{bs}$$

di mana *bs* adalah ukuran kumpulan gambar wajah, dan $I^{adv}{}_j$ adalah *adversarial image* ke-j dari sebuah *batch*. Operasi ini akan menyebabkan G_{avg} lebih berkonsentrasi pada atribut umum wajah manusia daripada atribut wajah tertentu. Kemudian, menggunakan PGD untuk menghasilkan adversarial perturbation P_{avg} melalui $G_{avg.}$

Setelah mendapatkan P_{avg} dari satu model, melakukan **fusi tingkat model**, yang secara iteratif menggabungkan P_{avg} yang dihasilkan dari model tertentu ke W_{CMUA} dalam pelatihan, dan W_{CMUA} awal hanyalah P_{avg} yang dihitung dari model deepfake pertama,

$$W_{CMUA}^{o} = p_{avg}^{0},$$

$$W_{CMUA}^{t+1} = \alpha \,.\, W_{CMUA}^{t} + (1-\alpha).\, P_{avg}^{t},$$

di mana α adalah faktor peluruhan, $P^{t}{}_{avg}$ adalah rata-rata gangguan yang dihasilkan dari model *deepfake* yang diserang ke-*t*, dan $W^{t}{}_{\text{CMUA}}$ adalah CMUA-*Watermark* pelatihan setelah model *deepfake* yang diserang ke-*t*

2.2.3.1.5 Automatic Step Size Tuning Based on TPE

Automatic Step Size Tuning Based on TPE (Tree-structured Parzen Estimator) adalah teknik yang digunakan untuk menentukan ukuran langkah atau step size yang optimal dalam algoritma optimisasi. TPE adalah metode yang mengestimasi distribusi probabilitas dari langkah-langkah yang memungkinkan dan menggabungkan estimasi tersebut untuk menghasilkan langkah-langkah yang

lebih baik pada iterasi selanjutnya (Bergstra et al., 2011). Dalam konteks watermarking, teknik ini dapat digunakan untuk menentukan langkah-langkah yang optimal dalam proses embedding atau pencarian watermark, sehingga meningkatkan efisiensi dan kualitas hasil watermarking.

Selain fusi dua tingkat yang disebutkan di atas, ditemukan bahwa ukuran langkah serangan untuk model yang berbeda juga penting untuk transferabilitas CMUA-Watermark yang dihasilkan. Oleh karena itu, mengeksploitasi pendekatan heuristik untuk secara otomatis menemukan ukuran langkah serangan yang sesuai.

Metode serangan dasar yang dipilih (PGD) termasuk ke dalam keluarga FGSM (Goodfellow et al., 2015), dan gradien $\nabla_x L$ dinormalisasi oleh fungsi *sign*:

$$sign\ x = \begin{cases} -1\ , & x < 0, \\ 0\ , & x = 0, \\ 1\ , & x > 0. \end{cases}$$

Dalam perhitungan nyata, elemen-elemen dalam $\nabla_x L$ hampir tidak pernah mencapai 0, sehingga $||\text{sign}(\nabla_x L)||2 \approx 1$ adalah tetap untuk setiap gradien. Perturbasi ΔP yang diperbarui dalam iterasi metode serangan berbasis *sign* dirumuskan sebagai:

$$\Delta P = a.sign(\nabla_X L).$$

Dengan kata lain, hanya ukuran langkah *a* yang menentukan tingkat pembaruan selama serangan, sehingga pemilihan *a* memiliki pengaruh yang besar terhadap performa serangan. Kesimpulan ini juga berlaku untuk serangan universal lintas model; perturbasi yang diperbarui ΔP^u dalam sebuah iterasi serangan universal lintas model dibentuk dengan menggabungkan ΔP^i dari beberapa model G_1,..., G_m:

$$\Delta P^{u} = \sum_{i=1}^{m} \alpha^{(m-i)} \Delta P_i = \sum_{i}^{m} \alpha^{(m-i)} a_i . sign(\nabla_X L_i) .$$

Dalam rumus di atas, *m* adalah jumlah model, faktor peluruhan α adalah sebuah konstanta, dan *sign* ($\nabla_X L_i$) memberikan arah optimasi untuk G_i . Oleh karena itu, arah optimasi secara keseluruhan sangat dipengaruhi oleh a_1, ..., a_m, dan memilih a_1, ..., a_m yang sesuai di berbagai model untuk menemukan arah keseluruhan yang ideal adalah masalah utama untuk serangan lintas model.

Menggunakan algoritma TPE (Bergstra et al., 2011) untuk memecahkan masalah ini, yang secara otomatis mencari a_1, ..., a_m yang sesuai untuk menyeimbangkan arah yang berbeda yang dihitung dari berbagai model. TPE adalah metode optimasi hiper-parameter berdasarkan *Sequential Model-Based Optimization* (SMBO), yang secara berurutan membangun model untuk memperkirakan kinerja hiper-parameter berdasarkan pengukuran historis, dan kemudian memilih hiperparameter baru untuk diuji berdasarkan model ini. Dalam penelitian ini, menganggap ukuran langkah a_1,...,a_m sebagai hyperparameter input *x* dan tingkat keberhasilan serangan sebagai nilai kualitas y dari TPE. TPE menggunakan $P(x|y)$ dan $P(y)$ untuk memodelkan $P(y|x)$, dan $p(x|y)$ diberikan oleh:

$$p(x|y) = \begin{cases} l(x), & if\ y < y^*, \\ g(x), & if\ y \geq y^*, \end{cases}$$

di mana y^* ditentukan oleh pengamatan terbaik secara historis, e(*x*) adalah densitas yang dibentuk dengan pengamatan $\{x(i)\}$ sedemikian rupa sehingga kerugian yang sesuai lebih rendah dari y^* , dan $g(x)$ adalah densitas yang dibentuk

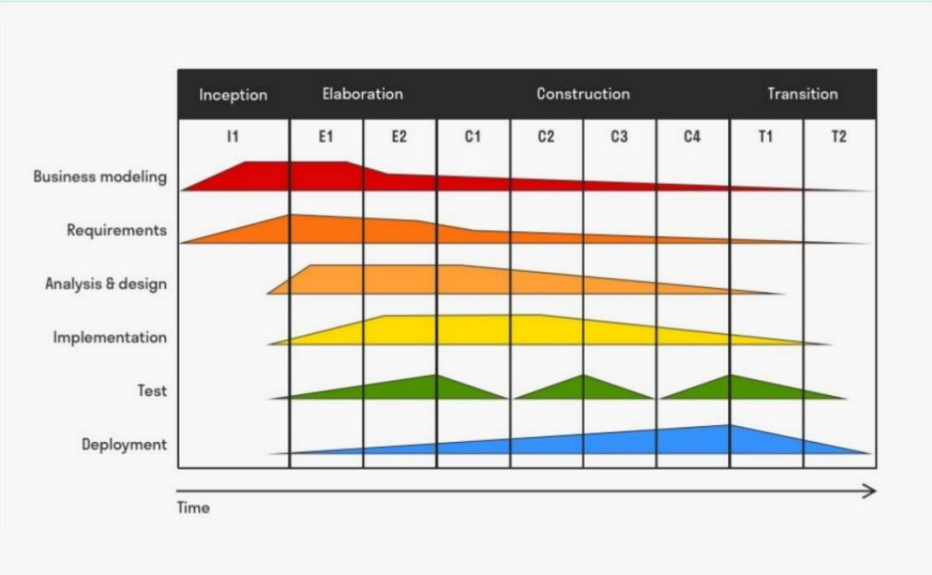
dengan pengamatan yang tersisa. Setelah memodelkan $P(y|x)$, lalu terus mencari ukuran langkah yang lebih baik dengan mengoptimalkan kriteria *Expected Improvement* (*EI*) di setiap iterasi pencarian, yang diberikan oleh,

$$EI_{y^*}(x) = \frac{\gamma y^* l(x) - l(x)\int_{-\infty}^{y^*} p(x)dy}{\gamma l(x) + (1-\gamma)g(x)}$$
$$\propto \left(y + \frac{g(x)}{l(x)}(1-\gamma)\right)^{-1},$$

di mana $\gamma = p(y < y^*)$. Dibandingkan dengan kriteria lainnya, *EI* bersifat intuitif dan telah terbukti memiliki kinerja yang sangat baik. Untuk detail lebih lanjut mengenai TPE, lihat (Bergstra et al., 2011).

2.2.4 Rational Unified Process

Rational Unified Process (RUP) adalah metode rekayasa pengembangan perangkat lunak yang digunakan untuk kedisiplinan dalam penetapan tugas dan tanggung jawab. Tujuan RUP adalah memastikan bahwa produk perangkat lunak yang dihasilkan akan berkualitas dan sesuai kebutuhan pengguna akhir (end-users) (Anwar, 2014).



Gambar II-2,Arsitektur Rasional Unified Process

RUP yang baik akan tercipta lewat hasil kerja sama antara pengembang perangkat lunak, mitra dan pengguna. Salah satu perspektif dalam RUP merupakan *Dynamic Perspective* & *Lifecycle Phases* yang penggunaannya digambarkan dalam bidang dua dimensi. Bidang horizontal menyatakan lamanya waktu pengembangan dan aspek dinamis lainnya, sedangkan bidang vertikal menyatakan aspek statis dalam rekayasa pengembangan perangkat lunak. Perspektif RUP model ini dinyatakan seperti dalam Gambar II-2.

Dalam bidang horizontal, terdapat fase atau tahap dalam proses rekayasa perangkat lunak yang memaparkan peran dari tiap unit. Fase dalam bidang ini terbagi ke dalam fase insepsi, elaborasi, konstruksi dan transisi.

1. Fase insepsi merupakan fase yang berfokus pada pendefinisian ruang lingkup atau batasan dalam proyek pengembangan dengan cara melakukan analisis desain berorientasi objek (*Object Oriented Analysis Design*).

Tujuan dari fase ini adalah untuk mendapatkan seluruh pemahaman dari pihak yang berkaitan agar sistem yang diajukan sesuai dengan keinginan dan kebutuhan.

2. Fase elaborasi merupakan fase yang akan membuat arsitektur dasar sistem lewat hasil analisis sebelumnya. Fase ini juga akan menentukan perencanaan proyek serta spesifikasi dari fitur yang akan dimuat dalam sistem. Hasil dari fase ini merupakan dokumen arsitektur yang berguna untuk fase selanjutnya.
3. Fase Konstruksi merupakan fase menerjemahkan spesifikasi fitur dari dokumen rancangan sebelumnya ke dalam bentuk program/sistem sesuai dengan arsitekturnya. Fase ini berfokus pada peningkatan fungsi serta implementasi yang lebih mendalam terhadap spesifikasi sistem.
4. Fase Transisi merupakan fase pengujian sistem ke pengguna akhir dimana sistem yang dibuat harus memenuhi kebutuhan perangkat lunak dan kebutuhan penggunanya. Kendali dalam fase ini mulai dipindah kepada tim pemeliharaan perangkat lunak.

2.3 Penelitian Lain yang Relevan

2.3.1 *Landmark Breaker: Obstructing DeepFake By Disturbing Landmark Extraction*

Penelitian yang telah dilakukan mengenai *Landmark Breaker: Obstructing DeepFake By Disturbing Landmark Extraction* (Sun et al., 2020). Tulisan ini menjelaskan metode baru, yaitu *Landmark Breaker*, untuk menghalangi generasi *DeepFake* dengan melanggar langkah prasyarat ekstraksi *landmark* wajah.

Dengan menciptakan *adversarial perturbations* untuk mengganggu ekstraksi *landmark* wajah, sehingga wajah input ke model *DeepFake* tidak dapat disejajarkan dengan baik. *Landmark Breaker* divalidasi pada himpunan data Celeb-DF, yang menunjukkan kemanjuran *Landmark Breaker* pada ekstraksi *landmark* wajah yang mengganggu.

2.3.2 Penelitian-Penelitian tentang *Face Modification*

Adapun penelitian tentang *face Modification* Dalam beberapa tahun terakhir, akses gratis ke gambar wajah berskala besar dan kemajuan luar biasa dari model generatif telah membuat jaringan modifikasi wajah menghasilkan gambar wajah yang lebih realistis dengan target orang atau atribut. StarGAN (Choi et al., 2018) mengusulkan pendekatan baru dan terukur untuk melakukan penerjemahan gambar-ke-gambar di berbagai domain, mencapai kualitas visual yang lebih baik pada gambar yang dihasilkan. Kemudian, AttGAN (He et al., 2019) menggunakan batasan klasifikasi atribut untuk memberikan gambar wajah yang lebih alami pada manipulasi atribut wajah. Selain itu, AGGAN (Tang et al., 2019) memperkenalkan *attention mask* melalui mekanisme perhatian bawaan untuk mendapatkan gambar target dengan kualitas tinggi. Baru-baru ini, (Li et al., 2021) mengusulkan HiSD yang merupakan metode penerjemahan gambar-ke-gambar yang canggih untuk skalabilitas beberapa label dan keragaman yang dapat dikontrol dengan pelepasan yang mengesankan. Meskipun model-model ini mengadopsi beragam arsitektur dan kerugian, *watermark* CMUA kami berhasil mencegah gambar wajah dimodifikasi dengan benar oleh semuanya.

2.4 Kesimpulan

Pada bab ini telah dibahas teori yang akan digunakan sebagai dasar penelitian ini. Pada bab ini juga telah dibahas mengenai penelitian terkait yang mendukung literatur penelitian ini. Mekanisme pelaksanaan penelitian selengkapnya akan dibahas dalam bab selanjutnya.

BAB III

METODE PENELITIAN

3.1 Pendahuluan

Pada bab ini akan dijelaskan mengenai tahapan penelitian, metode penelitian serta manajemen proyek penelitian. Tahapan penelitian dijadikan sebagai acuan pada setiap fase pengembangan perangkat lunak agar dapat memberikan solusi untuk rumusan masalah dan tercapainya tujuan penelitian.

3.2 Unit Penelitian

Penelitian ini dilaksanakan di Jurusan Teknik Informatika Fakultas Ilmu Komputer Universitas Sriwijaya.

3.3 Pengumpulan Data

Pada bagian ini akan dijelaskan tahapan pengumpulan data meliputi jenis dan sumber data dan metode pengumpulan data yang digunakan dalam penelitian.

3.3.1 Jenis Data

Jenis data yang digunakan sebagai objek penelitian ini ada dua yaitu, data primer dan sekunder.

3.3.2 Sumber Data

1. Data primer berupa kumpulan data citra foto KPM mahasiswa Fasilkom UNSRI Angkatan 2018.

2. data sekunder berupa *dataset* Celeb-A dari peneletian *Deep Learning Face Attributes in the Wild* (Liu et al., 2015).seperti pada Gambar III-1

Gambar III-1. Contoh data yang dari *dataset* celebA

3.3.3 Metode pengumpulan Data

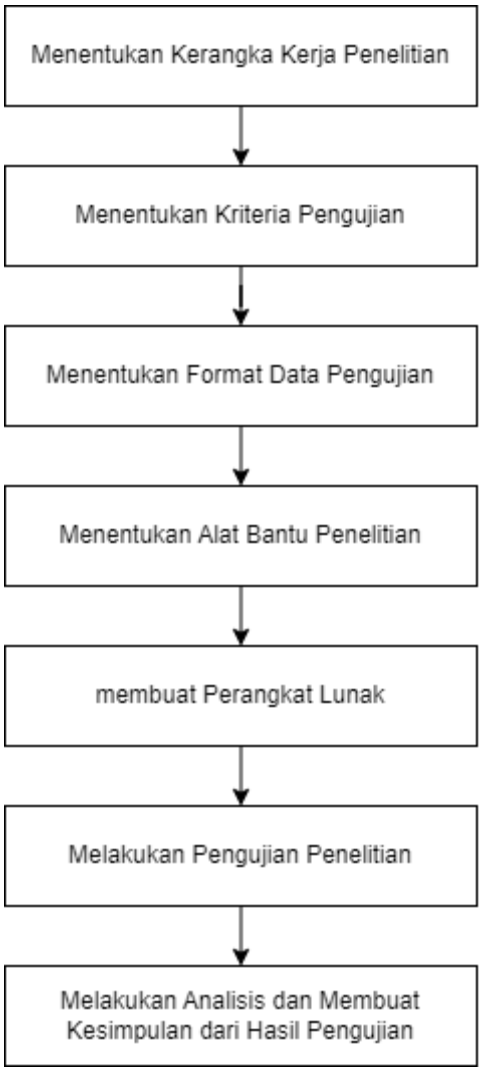
Ada dua metode pengumpulan data yang digunakan dalam penelitian ini:

1. *Crawling* data citra foto KPM mahasiswa Fasilkom Unsri Angkatan 2018 dari laman situs UNSRI lama[2].
2. mengunduh *dataset* Celeb-A dari di unduh dari halaman Kaggle[3].

[2] https://old.unsri.ac.id/?act=daftar_mahasiswa
[3] https://www.kaggle.com/datasets/nikhilbartwal001/celeba

3.4 Tahapan Penelitian

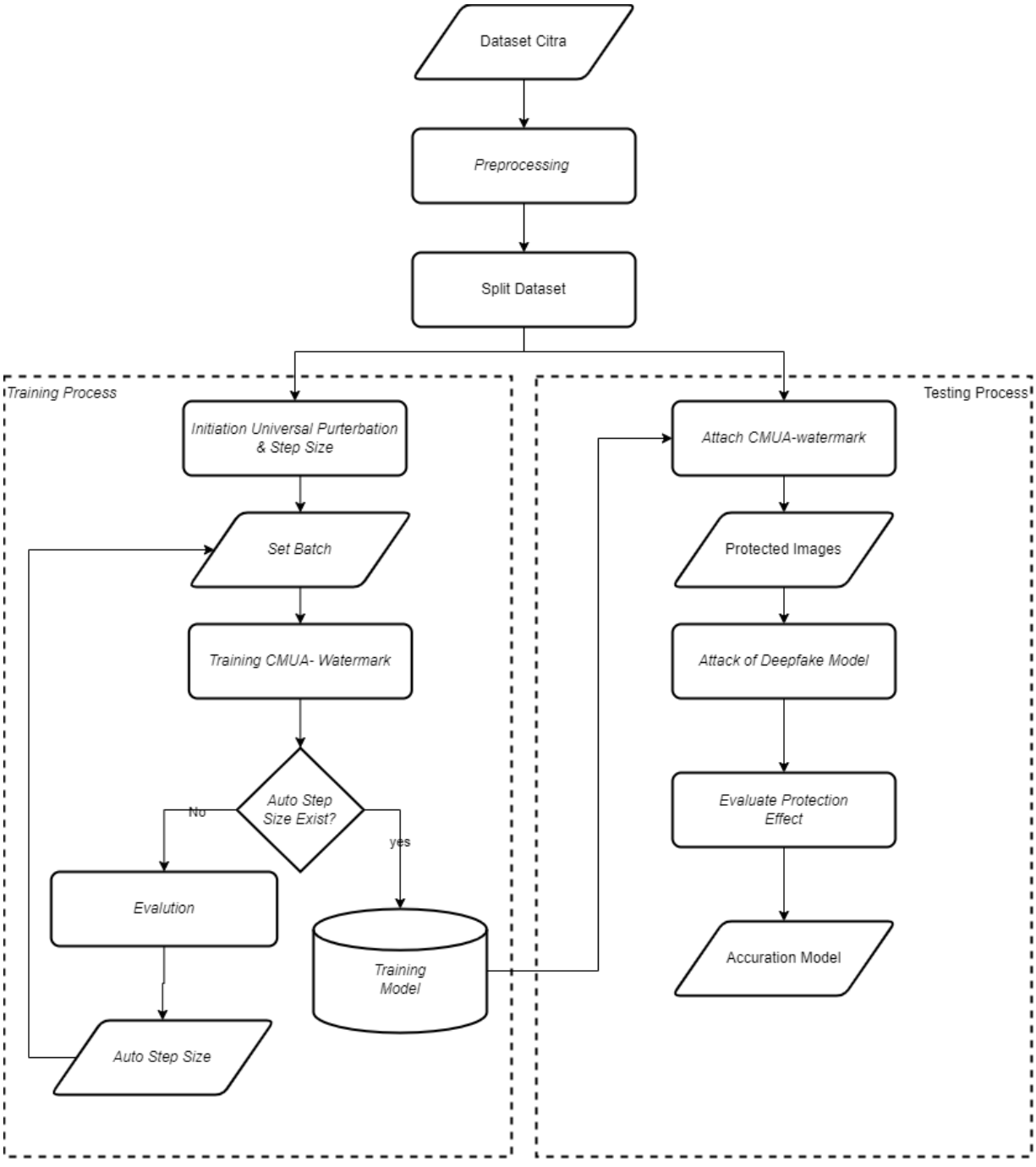
Tahapan penelitian adalah rincian proses yang akan dilakukan pada suatu penelitian. Tahapan penelitian yang akan dilakukan pada penelitian ini adalah sebagai berikut.



Gambar III-2. Alur Tahapan Penelitian

3.4.1 Kerangka Kerja

Kerangka kerja pada penelitian ini adalah sebagai berikut:



Gambar III-3. Diagram Alir Sistem Proteksi Citra dari *deepfake* dengan metode CMUA

Berdasarkan kerangka kerja penelitian pada Gambar III-2, sistem proteksi dari *deepfake* dengan metode CMUA memiliki alur sistem sebagai berikut:

1. ***Input Image Dataset***

   Pada proses ini memasukkan *dataset* yang digunakan dalam sistem ini, yaitu berupa *dataset* citra gambar wajah.

2. ***Proses Preprocessing***

   Pada proses ini melakukan restorasi kualitas citra gambar yang terdapat dalam dataset agar mudah di proses oleh sistem.

3. ***Split dataset***

   Membagi dataset menjadi dua bagian pertama digunakan untuk proses pelatihan, dan yang kedua digunakan sebagai proses validasi pada proses pengujian

4. ***Training Process***

   - Dimulai dengan melakukan pemasangan model proteksi awal yang bersifat acak serta memasangkan inisiasi awal step size dengan nilai - I_1~I_m untuk iterasi tiap serangan.
   - Input Jumlah Batch yang digunakan untuk proses pelatihan.
   - Melakukan proses training Model CMUA-*watermark*.
   - Memastikan apakah auto step size sudah ada atau belum didalam sistem,

     jika belum ada maka melanjukan proses evalusi untuk mendapatkan Auto step size yang akan digunakan untuk training di batch yang lebih besar,

Jika sudah ada proses dilanjutkan dengan menghasilkan model proteksi yang siap digunakan.

- Didapatkan Model proteksi yang CMUA-*watermark* yang siap digunakan.

5. Testing Process

- Data yang tidak melalui proses pelatihan akan dipasangankan dengan model proteksi yang sudah melalui proses pelatihan sebelumnya.
- Lalu menghasilkan output berupa gambar yang telah diproteksi.
- Kemudian kumpulan gambar yang telah diproteksi di serang dengan model deepfake
- Melakukan evaluasi efek dari gambar yang telah diproteksi
- Menghasilkan keluaran akhir berupa Akurasi tingkat berhasilan proteksi gambar dari deepfake

3.4.2 Kriteria Pengujian

Dalam melaksanakan pengujian penelitian ini, terdapat beberapa kriteria yang harus terpenuhi pada sistem yang dibangun. Adapun kriteria pengujian sebagai berikut:

1. Sistem dapat menggunakan metode CMUA-*Watermark* dalam memproteksi citra dari berbagai model *deepfake*.
2. Sistem dapat merima masukkan dari pengguna.
3. Sistem dapat menampilkan hasil tingkat keberhasilan proteksi citra gambar dari *deepfake.*

Selain kriteria tersebut, terdapat kriteria lainnya yang dapat dilihat secara lengkap pada bab IV.

3.4.3 Format data Pengujian

Format data pengujian yang digunakan berupa *table*

- L^2_{mask} adalah perbandingan antara *original image* dengan *distorted* image.
- SR_{mask} untuk merepresentasikan tingkat keberhasilan melindungi gambar wajah.
- FID mengukur kualitas pembuatan gambar wajah palsu.

Dataset	$L^2_{mask}\uparrow$	$SR_{mask}\uparrow$	FID↑
CelebA Training			
CelebA verifikasi			
Foto KPM mahasiswa Fasilkom Unsri 2018			

Tabel III-1. Tabel akurasi dan tingkat keberhasilan CMUA pada tiap *Dataset*

3.4.4 Alat yang digunakan dalam Pelaksanaan Penelitian

Alat bantu penelitian Proteksi Citra Foto KPM Mahasiswa Fasilkom UNSRI dari *DeepFake* dengan CMUA-*Watermark* yang akan digunakan dalam penelitian ini adalah sebagai berikut.

1. Perangkat Keras

   Proccessor : intel® core™ i7 8th *generation*

   RAM : 16GB RAM

   HDD : 1TB HDD strorage.

   VRAM : Nvidia Geforce GTX 1050 4GB RAM

2. Perangkat Lunak

   Sistem Operasi : Windows 10 64-bit

   Teks Editor : Visual Studio Code

   Basaha Pemprograman : Python

Selain perangkat lunak diatas diatas.terdapat tambahan perangkat lunak ketiga

3.4.5 Pengujian Penelitian

Dalam melakukan pengujian pada penelitian kali ini, pengujian akan dilakukan dengan memperhatikan indikator uji yang akan digunakan untuk mengetahui tingkat keakuratan dari sistem yang dibuat. Adapun indikator pengujian yang akan digunakan. Pengujian menggunakan matriks *Mask*, yang lebih berkonsentrasi pada area yang dimodifikasi,

$$Mask_{(i,j)} = \begin{cases} 1, & if \left|\left|G(I)_{(i,j)} - I_{(i,j)}\right|\right| > 0.5, \\ 0, & else \end{cases}$$

di mana (*i*, *j*) adalah koordinat piksel dalam gambar. Dengan cara ini, ketika menghitung L^2_{mask}, hanya piksel dengan perubahan besar yang akan dihitung dan area lainnya akan ditinggalkan,

$$L^2{}_{mask} = \frac{\sum_i \sum_j Mask_{(i,j)} . \left|\left|G(I)_{(i,j)} - G(I + C_{CMUA})_{(i,j)}\right|\right|}{\sum_i \sum_j Mask_{(i,j)}}.$$

Dalam eksperimen ini, jika $L^2_{mask} > 0{,}05$, kami menentukan bahwa gambar berhasil dilindungi, dan menggunakan SR_{mask} untuk merepresentasikan tingkat keberhasilan melindungi gambar wajah.

3.4.6 Analisis dan Kesimpulan Hasil Pengujian Penetian

Analisi hasil pengujian dilakukan dengan memperhatikan nilai $L^2_{mask\ untuk}$ menentukan bahwa gambar berhasil dilindungi dan SR_{mask} untuk merepresentasikan tingkat keberhasilan melindungi gambar wajah. Serta perbandingan tingkat keberhasilan dari masing masing *dataset* yang diujikan.

3.5 Metode Pengembangan Perangkat Lunak

Metode pengembangan yang digunakan dalam penelitian ini adalah metode Rational Unified Process (RUP). Pengembangan sistem deteksi kemiripan kode sumber dibagi ke dalam empat tahap, yaitu fase insepsi, fase elaborasi, fase konstruksi dan fase transisi. Berikut merupakan tahapan pengembangan perangkat lunak yang akan dilakukan dalam tiap fasenya.

3.5.1 Face Insepsi

Tahapan yang akan dilakukan dalam fase ini adalah sebagai berikut.

1. Pemodelan Sistem : Menentukan ruang lingkup dan batasan masalah.
2. Kebutuhan : Mendefinisikan spesifikasi perangkat lunak.
3. Analisis dan Perancangan : Melakukan analisis terhadap kebutuhan perangkat lunak termasuk di dalamnya kebutuhan fungsional dan non fungsional dari spesifikasi perangkat lunak.

4. Implementasi : Membuat seluruh rancangan sistem ke dalam bentuk diagram use-case.

3.5.2 Fase Elaborasi

Tahapan yang akan dilakukan dalam fase ini adalah sebagai berikut.

1. Pemodelan Sistem: Membuat rancangan antarmuka (interface) sistem.
2. Kebutuhan: Menentukan spesifikasi dari sistem.
3. Analisis dan Perancangan: Membangun model *activity diagram* dan *sequence diagram* dari rancangan sistem.
4. Impelementasi: Membuat program berdasarkan diagram yang ditentukan sebelumnya.

3.5.3 Fase Konstruksi

Tahapan yang akan dilakukan dalam fase ini adalah sebagai berikut.

1. Pemodelan Bisnis : Menentukan bahasa pemrograman yang akan membangun sistem.
2. Kebutuhan : Menentukan kebutuhan sistem sesuai dengan fungsi yang telah ditentukan.
3. Analisis dan Perancangan : Membangun tampilan antar-muka sistem.
4. Implementasi: Membangun sistem dengan membuat program menggunakan bahasa pemrograman yang telah ditentukan.

3.5.4 Fase Transisi

Tahapan yang akan dilakukan dalam fase ini adalah sebagai berikut.

1. Pemodelan Sistem : Menentukan pengujian terhadap sistem.

2. Kebutuhan : Menentukan alat bantu pengujian terhadap sistem.
3. Analisis dan Perancangan : Merancang kasus penggunaan selama pengujian sistem.
4. Implementasi : Melaksanakan pengujian terhadap sistem menggunakan kasus penggunaan yang telah ditentukan.

3.6 Manajemen Proyek Perangkat Lunak

Tujuan utama dari manajemen proyek penelitian yaitu melakukan perencanaan tentang aktivitas-aktivitas yang akan dilakukan selama proyek penelitian berlangsung. Aktivitas-aktivitas tersebut akan disusun dalam sebuah Work Breakdown Structure (WBS) yang tertera pada Tabel III-2.

ID	Work Point	Durasi (Hari)	Mulai	Selesai	*Predecessorr*
-	**Proteksi Citra Foto KPM Mahasiswa Fasilkom UNSRI dari DeepFake dengan CMUA-Watermark**	**120**	**6/11/22**	**5/3/2023**	
-	**Menentukan Ruang, Batasan, serta Unit Penelitian**	**6**	**6/11/22**	**12/11/22**	
W1	Pendalaman Mengenai Permasalahan	1	6/11/22	6/11/22	
W2	Analisa Permasalahan	1	7/11/22	7/11/22	W1
W3	Mencari Solusi Permasalahan	2	8/11/22	9/11/22	W2
W4	Membuat Dokumen Penelitian	3	10/11/22	12/11/22	W3
DP1	Tersedia Dokumen Penelitian	0	12/11/22	12/11/22	W4
-	**Menentukan Landasan Teori yang Berkaitan dengan Penyelesaian dan Tantangan dari Permasalahan**	**20**	**13/11/22**	**2/12/22**	
W5	Mengumpulkan Penelitian Terkait Baik Jurnal Maupun	5	13/11/22	17/11/22	

	Literatur Ilmiah Lainnya				
W6	Menganalisa Informasi Penelitian Terkait	6	18/11/22	23/11/22	W5
W7	Menentukan Solusi Pemecahan Permasalahan	4	24/11/22	27/11/22	W6
W8	Mempelajari Mengenai Solusi Terpilih	3	28/11/22	30/11/22	W7
W9	Membuat Dokumen Penelitian	2	1/12/22	2/12/22	W8
DP2	Tersedia Dokumen Penelitian	0	2/12/22	2/12/22	W9
-	**Menentukan Kriteria Pengujian**	**10**	**3/12/22**	**12/12/22**	
W10	Menentukan Format Data Yang Digunakan Dalam Pengujian	3	3/12/22	5/12/22	W7
W11	Menentukan Format Pengujian	3	6/12/22	8/12/22	W7
W12	Menentukan Format Analisa dan Kesimpulan Pengujian	3	9/12/22	11/12/22	W7
W13	Membuat Dokumen Penelitian	1	12/12/22	12/12/22	W10-W12
DP3	Tersedia Dokumen Penelitian		12/12/22	12/12/22	W13
-	**Fase Pembuatan Sistem**	**69**	**13/12/22**	**20/2/23**	
-	**Fase Insepsi**	**12**	**13/12/22**	**24/12/22**	
-	**Pemodelan Sistem**	**2**	**13/12/22**	**15/12/22**	
W14	Menentukan Ruang Lingkup dan Batasan Masalah	2	13/12/22	15/12/22	W2 & W7
-	**Kebutuhan**	**3**	**16/12/22**	**18/12/22**	
W15	Mengumpulkan Dataset Penelitian	3	16/12/22	18/12/22	W2
-	**Analisis dan Perancangan**	**3**	**19/12/22**	**21/12/22**	
W16	Membuat Diagram Alir Sistem	3	19/12/22	21/12/22	W14
-	**Implementasi**	**2**	**21/12/22**	**22/12/22**	
W17	Membuat Dokumen dari Perancangan Sistem	2	21/12/22	22/12/22	W14
-	**Pengujian**	**2**	**23/12/22**	**24/12/22**	
W18	Melakukan Validasi Fungsionalitas Sistem	2	23/12/22	24/12/22	W17

-	**Fase Elaborasi**	**15**	**25/12/22**	**8/1/23**	
-	**Pemodelan Sistem**	**5**	**25/12/22**	**29/12/22**	
W19	Membuat Arsitektur Perangkat Lunak Berdasarkan Fungsionalitas Sistem	5	25/12/22	29/12/22	W14
-	**Kebutuhan**	**4**	**30/12/22**	**2/1/23**	
W20	Melengkapi Dataset Penelitian	4	30/12/22	2/1/23	W15
-	**Analisis dan Perancangan**	**2**	**3/1/23**	**4/1/23**	
W21	Membuat Dokumentasi Berupa *Activity* dan *Sequence* Diagram	2	3/1/23	4/1/23	W14 & W16
-	**Implementasi**	**2**	**5/1/23**	**6/1/23**	
W22	Melengkapi Dokumen Penelitian	2	5/1/23	6/1/23	W21
-	**Pengujian**	**2**	**7/1/23**	**8/1/23**	
W23	Melakukan Validasi Terhadap Arsitektur yang Ditentukan	2	7/1/23	8/1/23	W19
-	**Fase Konstruksi**	**25**	**9/1/23**	**3/2/23**	
-	**Pemodelan Sistem**	**10**	**9/1/23**	**18/1/23**	
W24	Membuat Tampilan Antarmuka Berdasarkan Rancangan Sistem	10	9/1/23	18/1/23	W19
-	**Kebutuhan**	**2**	**19/1/23**	**20/1/23**	
W25	Menentukan Bahasa Pemrograman dan Perangkat Keras Yang Digunakan	2	19/1/23	20/1/23	W19
-	**Analisis dan Perancangan**	**3**	**21/1/23**	**23/1/23**	
W26	Membuat Dokumentasi Berupa *Class Diagram*	3	21/1/23	23/1/23	W21
-	**Implementasi**	**10**	**24/1/23**	**3/2/23**	
W27	Melakukan Implementasi *Class Diagram* Kedalam Sistem	10	24/1/23	3/2/23	W26
-	**Fase Transisi**	**15**	**4/2/23**	**18/2/23**	
-	**Pemodelan Sistem**	**3**	**4/2/23**	**7/2/23**	
W28	Membuat Skenario Pengujian Sistem	3	4/2/23	7/2/23	W27
-	**Kebutuhan**	**3**	**8/2/23**	**10/2/23**	
W29	Menentukan Data Yang Digunakan Dalam Pengujian	3	8/2/23	10/2/23	W28

	Sistem				
-	**Analisis dan Perancangan**	**3**	**11/2/23**	**13/2/23**	
W30	Membuat Dokumentasi Tabel Pengujian	3	11/2/23	13/2/23	W28 & W29
-	**Implementasi**	**3**	**14/2/23**	**16/2/23**	
W31	Melakukan Pengujian Sistem Sesuai Skenario Yang Ditentukan	3	14/2/23	16/2/23	W29 – W30
-	**Pengujian**	**3**	**17/2/23**	**19/2/23**	
W32	Meninjau Hasil Pengujian Berdasarkan Skema Analisis Yang Ditentukan	3	17/2/23	19/2/23	W31
DP4	Tersedia Dokumen Penelitian		19/2/23	19/2/23	W13
-	**Melakukan Analisis Hasil Pengujian Sistem**	**15**	**19/2/23**	**5/3/2023**	
W33	Melakukan Analisis Hasil Berdasarkan Keluaran Dari Pengujian Sistem	9	21/2/23	27/2/23	W32
W34	Menarik kesimpulan berdasarkan analisis hasil pengujian	6	28/2/23	5/3/2023	W33
DP5	Tersedia Dokumen Penelitian		5/3/2023	5/3/2023	W34

Tabel III-2. Work Breakdown Structure (WBS) Pembuatan Sistem

3.7 Kesimpulan

Pada bab ini telah dibahas tentang proses pengumpulan data yang digunakan sebagai bahan uji perangkat lunak, tahapan penelitian, metode pengembangan perangkat lunak yang akan digunakan serta kriteria pengujian penelitian yang akan dilakukan terhadap sistem.

DAFTAR PUSTAKA

Anwar, A. (2014). A Review of RUP (Rational Unified Process). *International Journal of Software Engineering*, *5*(2), 8–24. http://www.cscjournals.org/library/manuscriptinfo.php?mc=IJSE-142

Bergstra, J., Bardenet, R., Bengio, Y., & Kégl, B. (2011). Algorithms for hyper-parameter optimization. *Advances in Neural Information Processing Systems 24: 25th Annual Conference on Neural Information Processing Systems 2011, NIPS 2011*, 1–9.

Choi, Y., Choi, M., Kim, M., Ha, J. W., Kim, S., & Choo, J. (2018). StarGAN: Unified Generative Adversarial Networks for Multi-domain Image-to-Image Translation. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 8789–8797. https://doi.org/10.1109/CVPR.2018.00916

Day, C. (2019). The Future of Misinformation. *Computing in Science and Engineering*, *21*(1), 108. https://doi.org/10.1109/MCSE.2018.2874117

Fletcher, J. (2018). Deepfakes, artificial intelligence, and some kind of dystopia: The new faces of online post-fact performance. *Theatre Journal*, *70*(4), 455–471. https://doi.org/10.1353/tj.2018.0097

Gonzalez, R. C., & Woods, R. E. (2018). *4TH EDITION Digital image processing*.

Goodfellow, I. J., Shlens, J., & Szegedy, C. (2015). Explaining and harnessing adversarial examples. *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, 1–11.

He, Z., Zuo, W., Kan, M., Shan, S., & Chen, X. (2019). AttGAN: Facial Attribute Editing by only Changing What You Want. *IEEE Transactions on Image Processing*, *28*(11), 5464–5478. https://doi.org/10.1109/TIP.2019.2916751

Huang, H., Wang, Y., Chen, Z., Li, Y., Tang, Z., Chu, W., Chen, J., Lin, W., & Ma, K.-K. (2021). CMUA-*Watermark: A Cross-Model Universal Adversarial Watermark for Combating Deepfakes*. http://arxiv.org/abs/2105.10872

Jain, P., Dave, M., & Patel, V. M. (2020). A Comprehensive Review on Steganography Techniques in Digital Images. *Journal of King Saud University-Computer and Information Sciences*, *32*(4), 395–408.

Korshunov, P., & Marcel, S. (2018). *DeepFakes: a New Threat to Face Recognition? Assessment and Detection*. 1–5. http://arxiv.org/abs/1812.08685

Li, X., Zhang, S., Hu, J., Cao, L., Hong, X., Mao, X., Huang, F., Wu, Y., & Ji, R. (2021). Image-to-image Translation via Hierarchical Style Disentanglement. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, *i*, 8635–8644. https://doi.org/10.1109/CVPR46437.2021.00853

Liu, Z., Luo, P., Wang, X., & Tang, X. (2015). Deep learning face attributes in the wild. *Proceedings of the IEEE International Conference on Computer Vision*, *2015 Inter*, 3730–3738. https://doi.org/10.1109/ICCV.2015.425

M. Sonka, V. H. and R. B. (2014). Image processing, analysis, and machine vision. Cengage Learning. In *IEEE Aerospace and Electronic Systems Magazine*.

Madry, A., Makelov, A., Schmidt, L., Tsipras, D., & Vladu, A. (2018). Towards deep learning models resistant to adversarial attacks. *6th International Conference on Learning Representations, ICLR 2018 - Conference Track Proceedings*, 1–28.

Rafique, M. A., Younus, S., & Bhatti, M. A. (2018). A comprehensive review on digital image representation and its applications. *Digital Communications and Networks*, *4*(1), 1–14.

Ruiz, N., Bargal, S. A., & Sclaroff, S. (2020). Disrupting Deepfakes: Adversarial Attacks Against Conditional Image Translation Networks and Facial Manipulation Systems. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *12538 LNCS*, 236–251. https://doi.org/10.1007/978-3-030-66823-5_14

Sun, P., Li, Y., Qi, H., & Lyu, S. (2020). Landmark Breaker: Obstructing DeepFake by Disturbing Landmark Extraction. *2020 IEEE International*

Workshop on Information Forensics and Security, WIFS 2020, 6–11. https://doi.org/10.1109/WIFS49906.2020.9360910

Tang, H., Xu, D., Sebe, N., & Yan, Y. (2019). Attention-Guided Generative Adversarial Networks for Unsupervised Image-to-Image Translation. *Proceedings of the International Joint Conference on Neural Networks*, *2019-July*. https://doi.org/10.1109/IJCNN.2019.8851881

Tolosana, R., Vera-Rodriguez, R., Fierrez, J., Morales, A., & Ortega-Garcia, J. (2020). Deepfakes and beyond: A Survey of face manipulation and fake detection. *Information Fusion*, *64*, 131–148. https://doi.org/10.1016/j.inffus.2020.06.014