

SKRIPSI

ANALISIS KEASLIAN FOTO DIGITAL MENGGUNAKAN *ERROR LEVEL ANALYSIS* (ELA) DAN *SCALE-INVARIANT FEATURE TRANSFORM* (SIFT)



OLEH
Nirwandi Bahri
07352011022

PROGRAM STUDI INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS KHAIRUN
TERNATE
2024

LEMBAR PENGESAHAN

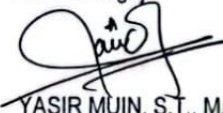
ANALISIS KEASLIAN FOTO DIGITAL MENGGUNAKAN *ERROR LEVEL ANALYSIS* (ELA) DAN *SCALE-INVARIANT FEATURE TRANSFORM* (SIFT)

Oleh
Nirwandi Bahri
07352011022

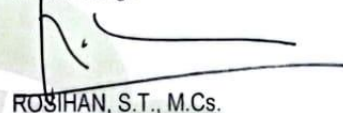
Skripsi ini telah disahkan
Tanggal 10 Juli 2024

Menyetujui
Tim Penguji

Ketua Penguji


YASIR MUIN, S.T., M.Kom.
NIDN. 9990582796

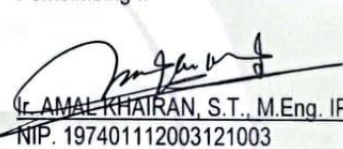
Pembimbing I


ROSIHAN, S.T., M.Cs.
NIP. 197607192010121001

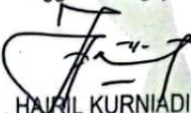
Anggota Penguji


ALFANUGRAHA, HI. USMAN, S.T., M.Kom.
NIP. 199403182019032029

Pembimbing II


Ir. AMAL KHAIRAN, S.T., M.Eng. IPM.
NIP. 197401112003121003

Anggota Penguji


HAURIL KURNIADI SIRAJUDDIN, S.Kom., M. Kom.
NIP. 198204272023211009

Koordinator Program Studi
Informatika


ROSIHAN, S.T., M.Cs.
NIP. 197607192010121001

Mengetahui/Menyetujui


Ir. ENDAH HARISUN, S.T., M.T., CRP.
NIP. 197511302006011013
Dekan Fakultas Teknik
Universitas Khairun

LEMBAR PERNYATAAN KEASLIAN

Yang bertanda tangan di bawah ini:

Nama : Nirwandi Bahri
NPM : 07352011022
Fakultas : Teknik
Jurusan/Program Studi : Informatika
Judul Skripsi : Analisis Keaslian Foto Digital Menggunakan *Error Level Analysis* (ELA) dan *Scale-Invariant Feature Transform* (SIFT)

Dengan ini menyatakan bahwa penulisan Skripsi yang telah saya buat ini merupakan hasil karya sendiri dan benar keasliannya. Apabila ternyata di kemudian hari penulisan Skripsi ini merupakan hasil plagiat terhadap karya orang lain, maka saya bersedia mempertanggung jawabkan sekaligus bersedia menerima sanksi berdasarkan aturan tata tertib di Universitas Khairun.

Demikian pernyataan ini saya buat dalam keadaan sadar dan tidak dipaksakan.

Penulis

Nirwandi Bahri



HALAMAN PERSEMBAHAN

Dengan Rahmat Allah *Subhanahu Wa Ta'ala* yang Maha Pengasih lagi Maha Penyayang serta mengucapkan rasa syukur *Alhamdulillah* atas nikmat yang di berikan tanpa hentinya, penulis persembahkan skripsi ini kepada:

1. Bahri Muhammad dan Nur Intan Ngewa, kedua orang tua hebat yang telah senantiasa memberikan kasih sayang, nasihat, motivasi dan dukungan, serta doa kepada penulis sehingga dapat menyelesaikan penulisan skripsi ini. Semoga beliau selalu bangga kepada penulis. Terima kasih mama dan papa.
2. Terimakasih kepada saudara dan saudari saya yang bernama Nurmiyati Bahri, Nadia Tri Utami Bahri, Nirwansya Bahri dan Nurfadila Naumi Bahri, serta keluarga (Besar Ngewa) yang telah memberikan dukungan kepada saya, sehingga dapat menyelesaikan tugas akhir saya ini (Skripsi)
3. Terima kasih kepada keluarga besar saya yang selalu memberikan dukungan dan dorongan tanpa henti sepanjang perjalanan ini.
4. Terima kasih untuk teman-teman Panda Squad yaitu Suci Ayu Maharani, Sasmita Hi. Sadek, Lisa Elisia Potale, Nurwana Iswan, Aprilia Silawane, Harlina Sapsuha, Nafra Aziqra Hi. A, Marhama Maynaka, Cindy Rahawaty S. Hipy, Muhammad Raihan Rizal, Nirwandi Bahri, Ferliy, dan Rinaldi Abdul karena telah membantu dan menyemangati saya dalam menyelesaikan skripsi ini.
5. Terima kasih kepada dosen dosen yang telah membantu dan mengarahkan saya dalam penulisan skripsi ini sehingga dapat menyelesaikan skripsi ini.
6. Terima kasih kepada kak Satria Dwi Surya S.T., M.Kom. sebagai kakak dan sekaligus pembimbing 3 saya yang selalu membantu dan mengarahkan saya selama ini.
7. Terima kasih kepada diri sendiri yang selalu semangat untuk menyelesaikan skripsi ini. Terima kasih telah mengandalkan diri sendiri untuk tetap kuat tanpa bergantung kepada orang lain dan tidak pernah memutuskan untuk menyerah dalam situasi apapun.

MOTTO

“Jalan Saja, Nanti Juga Sampai”

KATA PENGANTAR

Puji syukur saya panjatkan kepada Allah *Subhanahu Wata'ala* yang telah melimpahkan rahmat, taufik dan hidayah-Nya sehingga penulis dapat menyusun skripsi ini dengan judul “Analisis Keaslian Foto Digital Menggunakan *Error Level Analysis* (ELA) dan *scale-invariant feature transform* (SIFT)”, ini dapat diselesaikan untuk memenuhi salah satu persyaratan penyelesaian pendidikan sarjana Informatika Strata Satu (S1) pada Program Studi Informatika Fakultas Teknik Universitas Unkhair.

Untuk menyelesaikan hasil ini penulis sepenuhnya mendapat dukungan dari banyak pihak, oleh karena itu dengan rendah hati penulis mengucapkan terimakasih kepada:

1. Bapak Dr. Ridha Ajam, M.Hum., selaku Rektor Universitas Khairun Ternate.
2. Bapak Ir. Endah Harisun, S.T., M.T., CRP., selaku dekan Fakultas Teknik Universitas Khairun.
3. Bapak Rosihan, S.T., M.Cs., selaku Koordinator Progran Studi Informatika Fakultas Teknik Universitas Khairun dan sekaligus sebagai Pembimbing I, terima kasih atas bimbingannya, serta dukungan dalam penyelesaian hasil ini.
4. Bapak Ir. Amal Khairan, S.T., M.Eng., IPM., sebagai Pembimbing II, terima kasih atas bimbingannya, serta dukungan dalam penyelesaian hasil ini.
5. Bapak Yasir Muin, S.T., M.Kom., sebagai Penguji I, terima kasih atas bimbingannya, serta dukungan dalam penyelesaian hasil ini.
6. Ibu Alfanugrah A. Hi. Usman, S.T., M.Kom, sebagai Penguji II, terima kasih atas bimbingannya, serta dukungan dalam penyelesaian hasil ini.
7. Bapak Ir. Abdul Mubarak, S.Kom., M.T., IPM., sebagai Penguji III, terima kasih atas bimbingannya, serta dukungan dalam penyelesaian hasil ini.
8. Bapak Hairil Kurniadi Sirajuddin, S.Kom., M.Kom., selaku Penguji III, yang telah bersedia meluangkan waktu untuk menguji dan memberikan masukan perbaikan demi menyempurnakan Skripsi ini.
9. Kepada bapak saya yang bernama Bahri Muhammad yang telah memberikan dukungan dan kasih sayang dan senantiasa dilimpahkan kepada penulis hingga saat ini.
10. Kepada mama saya yang bernama Nur Intan Ngewa yang telah memberikan dukungan dan kasih sayang dan senantiasa dilimpahkan kepada penulis hingga saat ini.

ini.

11. Dan kepada saudara dan saudari saya yang bernama Nurmiyati Bahri, Nadia Tri Utami Bahri, Nirwansya Bahri dan Nurfadila Naumi Bahri, serta keluarga (Besar Ngewa) yang telah memberikan dukungan kepada saya, sehingga dapat menyelesaikan tugas akhir saya ini (Skripsi)
12. Kepada teman-teman seperjuangan (Panda Squad) diantaranya yaitu Suci Ayu Maharani, Sasmita Hi. Sadek, Lisa Elisia Potale, Nurwana Iswan, Aprilia Silawane, Harlina Sapsuha, Nafra Aziqra Hi. A, Marhama Maynaka, Wahyudin Nurdin, Muhammad Raihan Rizal, Ferliy, dan Rinaldi Abdul atas dukungan dan kerjasamanya selama menempuh pendidikan serta penyelesaian penyusunan skripsi ini.
13. Dan kepada diri saya Nirwandi Bahri saya ucapkan banyak terimakasih karena mampu bertahan sampai titik ini.

Walaupun demikian dalam hasil ini, penulis menyadari masih belum sempurna. oleh karena itu harapan penulis dalam memberikan kritik dan saran dari semua pihak yang bersifat membangun.

Ternate 13 Febuari 2024

Penulis

DAFTAR ISI

	Halaman
HALAMAN JUDUL	i
HALAMAN PENGESAHAN	ii
HALAMAN PERNYATAAN KEASLIAN	iii
HALAMAN PERSEMBAHAN.....	iv
KATA PENGANTAR	v
DAFTAR ISI	vii
DAFTAR GAMBAR	x
DAFTAR TABEL	xii
ABSTRAK.....	xiii
 BAB I PENDAHULUAN	
1.1. Latar Belakang	1
1.2. Rumusan Masalah	3
1.3. Batasan Masalah.....	4
1.4. Tujuan Penelitian.....	4
1.5. Manfaat Penelitian.....	4
1.6. Sistematika Penulisan	5
 BAB II TINJAUAN PUSTAKA	
2.1. Penelitian Terkait.....	6
2.2. Analisis	10
2.3. Foto.....	10
2.4. Citra	11
2.5. Analisis Citra.....	11
2.6. <i>Error Level Ananlysis (ELA)</i>	12
2.7. <i>Scale-Invariant Feature Transform (SIFT)</i>	13
2.8. <i>Flowchart</i>	14
2.9. <i>Python</i>	16
2.10. <i>Library</i>	16
2.11. <i>White Box</i>	16

2.12.	<i>Agile</i>	17
2.13.	Model <i>Agile</i>	19
2.14.	Metode Penelitian	19
2.14.1.	Penelitian Kualitatif	20
2.14.2.	Penelitian Kualitatif	20
2.14.3.	Penelitian dan Pengembangan	20

BAB III METODE PENELITIAN

3.1	Alur Penelitian	22
3.2.	Objek dan Waktu Penelitian	22
3.3.	Metode Pengumpulan Data.....	23
3.4.	Alat dan Bahan Penelitian	24
3.4.1.	Perangkat Keras (<i>Hardware</i>).....	24
3.4.2.	Perangkat Lunak (<i>Software</i>).....	24
3.5.	Metode Pengembangan Sistem.....	25
3.5.1.	<i>Requirements</i> (Kebutuhan)	25
3.5.2.	<i>Planning</i> (Perencanaan).....	27
3.5.3.	<i>Desing</i> (Desain).....	28
3.5.4.	<i>Implementation</i> (Implementasi).....	30
3.5.5.	<i>System Testing</i> (Pengujian Sistem)	31
3.6.	Impementasi <i>Library</i>	31
3.7.	Contoh Kasus	32
3.8.	Perancangan Pengujian Sistem	32

BAB IV HASIL DAN PEMBAHASAN

4.1.	Analisis Data.....	34
4.2.	Implementasi ELA dan SIFT	34
4.3.	Implementasi Sistem	39
4.4.	Menghubungkan API Dengan Sistem	44
4.5.	Hasil Analisis	44
4.6.	<i>White Box</i>	50

BAB V PENUTUP

5.1.	Kesimpulan.....	55
------	-----------------	----

5.2. Saran	55
------------------	----

DAFTAR PUSTAKA

LAMPIRAN

DAFTAR GAMBAR

	Halaman
Gambar 2.1 Contoh cara kerja ELA (Sudianto 2023).....	13
Gambar 2.2 Contoh cara kerja metode SIFT (Khairan 2023)	14
Gambar 2.3. Meteologi <i>Agile</i> (Nova,2022)	19
Gambar 3.1 Alur Penelitian.....	22
Gambar 3.2 Model <i>Agile</i>	25
Gambar 3.3 Perancangan Arsitektur	28
Gambar 3.4 Tampilan Awal Aplikasi.....	29
Gambar 3.5 Tampilan Utama Aplikasi.....	29
Gambar 3.6 Tampilan Hasil Foto Asli	30
Gambar 3.7 Tampilan Hasil Foto Manipulasi	30
Gambar 4.1. Dataset Foto Asli dan Manipulasi	34
Gambar 4.2. <i>Import Library</i>	35
Gambar 4.3. Konfigurasi Kelas (<i>Class Config</i>).....	35
Gambar 4.4. Fungsi Computer ELA Cv.....	36
Gambar 4.5. Fungsi ELA	36
Gambar 4.6. Fungsi Main	37
Gambar 4.7. Eksekusi Program.....	37
Gambar 4.8. Analisis Foto Menggunakan ELA	38
Gambar 4.9. Analisis Foto Menggunakan SIFT	39
Gambar 4.10. Tampilan Awal Aplikasi.....	40
Gambar 4.11. Tampilan Utama Aplikasi.....	41
Gambar 4.12. Tampilan Hasil Foto Asli.....	42
Gambar 4.13. Tampilan Hasil Foto Manipulasi	43
Gambar 4.14 Menghubungkan API Dengan Sistem	44
Gambar 4.15. Proses epoch ke-10.....	45
Gambar 4.16. Kurva Akurasi dan kurva <i>Loss</i>	47
Gambar 4.17. Proses epoch ke-24.....	48
Gambar 4.18. Proses epoch ke-30.....	49

Gambar 4.19. <i>Flowchart</i> Data Prediksi.....	51
Gambar 4.20. <i>Flowgrap</i> Pengujian Data Prediksi	52

DAFTAR TABEL

	Halaman
Tabel 2.1. Penelitian Terkait	6
Tabel 2.2. Contoh Nilai ELA Per 8x8 pixel (Candra,2020)	12
Tabel 2.3. Simbol dan Fungsi <i>Flowchat</i> (Charis Noiya,2023)	15
Tabel 3.1. Detail Spesifikasi <i>Hardware</i>	24
Tabel 3.2. Detail Spesifikasi <i>Software</i>	24
Tabel 3.3. Penerapan <i>Library</i>	24
Tabel 4.1. Implementasi penerapan <i>Error Level Analysis</i> (ELA)	31
Tabel 4.2. Epoch 1/10	37
Tabel 4.3. Epoch 10/10	45
Tabel 4.4. Epoch 1/24	47
Tabel 4.5. Epoch 24/24	47
Tabel 4.6. Epoch 1/30	49
Tabel 4.7. Epoch 30/30	49

ABSTRAK

ANALISIS KEASLIAN FOTO DIGITAL MENGGUNAKAN *ERROR LEVEL ANALYSIS* (ELA) DAN *SCALE-INVARIANT FEATURE TRANSFORM* (SIFT)

Nirwandi Bahri¹, Rosihan², Amal Khairan³

Program studi Informatika, Fakultas Teknik, Universitas Khairun

Jl. Jati Metro, Kota Ternate Selatan

Email: nirwandi221@gmail.com¹, rosihan@unkhair.ac.id², 4malkhairan@unkhair.ac.id³

Perkembangan teknologi informasi yang semakin pesat memungkinkan semua kegiatan masyarakat dapat dilakukan secara digital. Salah satu kegiatan masyarakat yang sering dilakukan di era digital ini adalah melakukan image editing. *Image editing* di era digital ini dilakukan dengan menggunakan *software editing tools* sehingga foto hasil editan tersebut terlihat seperti nyata. Jenis manipulasi pada image editing bisa bermacam-macam baik itu menambahkan *background*, menambah atau mengurangi kecerahan foto, menambahkan objek tertentu, memanipulasi bagian tertentu, ataupun hanya sekedar melakukan *cropping* (pemotongan gambar). Masalah yang dibahas adalah tantangan dalam mendeteksi modifikasi pada gambar digital khususnya mengidentifikasi metode yang efektif untuk tujuan ini. Penelitian ini bertujuan untuk menerapkan dan menganalisis efektivitas algoritma Analisis Tingkat Kesalahan (ELA) dan Transformasi Fitur Invarian Skala (SIFT) dalam mendeteksi manipulasi foto. Hasil penelitian menunjukkan bahwa menggabungkan ELA dan SIFT memberikan pendekatan yang kuat untuk mengidentifikasi pemalsuan gambar digital meningkatkan pemahaman dan alat yang tersedia untuk forensik foto digital. Dengan penelitian ini, diharapkan dapat meningkatkan pemahaman masyarakat tentang keaslian suatu foto digital sehingga mereka dapat lebih bijak dalam menilai dan membagikan informasi melalui platform-platform digital. Sistematis penulisan skripsi ini merupakan pembahasan singkat dari setiap bab yang menjelaskan hubungan antara bab satu dengan bab yang lainnya, yaitu latar belakang, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, serta sistematis penulisan. Dengan menggunakan metode ELA dan SIFT, penelitian ini memberikan kontribusi signifikan dalam bidang keamanan digital dan pengenalan manipulasi foto, yang penting dalam konteks media sosial dan jurnalisme.

Kata Kunci: Forensik Gambar Digital, Analisis Tingkat Kesalahan (ELA), Transformasi Fitur Invarian Skala (SIFT), Deteksi Pemalsuan *Copy-Move*, Manipulasi Foto, Keaslian Gambar

BAB I

PENDAHULUAN

1.1. Latar belakang

Perkembangan teknologi informasi yang semakin pesat memungkinkan semua kegiatan masyarakat dapat dilakukan secara digital. Meskipun demikian, salah satu kegiatan masyarakat yang sering dilakukan di era digital ini adalah melakukan *image editing*. *Image editing* di era digital ini dilakukan dengan menggunakan *software editing tools*, sehingga foto hasil editan tersebut terlihat seperti nyata. Jenis manipulasi pada *image editing* bisa bermacam-macam, baik itu menambahkan *background*, menambah atau mengurangi kecerahan foto, menambahkan objek tertentu, memanipulasi bagian tertentu, ataupun hanya sekedar melakukan *cropping* (pemotongan gambar). Banyak aplikasi yang menawarkan kemudahan untuk mengolah citra. Dalam memudahkan pengolahan citra digital, yang memanfaatkan teknologi tersebut serta memiliki dampak negatif. Adanya citra digital yang dimodifikasi dapat memunculkan kasus sosial semacam penyebaran data yang tidak benar, sehingga sangat mudah sekali terjadi kesalah pahaman antara bermacam pihak (Wicaksono, 2022).

Meskipun saat ini manipulasi citra sering disalahgunakan oleh segelintir orang untuk digunakan sebagai media dalam membuat foto palsu atau penipuan, salah satu jenis pemalsuan citra yang paling umum dan paling sering digunakan oleh seseorang adalah dengan melakukan teknik manipulasi gambar *copy-move*. Teknik *copy-move* sering digunakan karena sangat mudah dan efektif untuk dilakukan. *Copy-move* merupakan teknik pemalsuan citra digital dengan cara membuat suatu objek menghilang dari foto dan memindahkannya ke gambar yang lain. Dengan banyaknya kejahatan yang dilakukan

melalui manipulasi citra ini, diperlukan sebuah mekanisme khusus untuk menganalisis dan menelusuri bukti-bukti digital yang ada (Hari, 2020).

Kasus manipulasi citra yang terjadi saat ini, misalnya, adalah beredarnya foto-foto palsu yang sensitif sehingga menimbulkan informasi palsu kepada masyarakat, sehingga menimbulkan permasalahan-permasalahan dalam kehidupan sosial, seperti penyebaran informasi-informasi yang tidak benar, hingga sangat mudah terjadi kesalah pahaman, seperti penipuan online. Khususnya, pemalsuan citra digital sering terjadi dalam konteks kejahatan sosial. Identifikasi pemalsuan citra dalam distribusi digital sulit karena citra asli yang dimodifikasi sulit dikenali. Motivasi manipulasi citra bervariasi, mulai dari hiburan hingga tujuan pemasaran dan pengancaman, termasuk upaya memperkuat penyelidikan (Purnama, 2023).

Meyakinkan keaslian dan otentikasi tanpa keraguan (*non-repudiation*) memerlukan metode yang efektif. Salah satu metode yang baik untuk mendeteksi manipulasi pada foto digital menggunakan *Error Level Analysis* (ELA). Area citra yang dicurigai terekayasa akan menghasilkan sebaran *error level* yang berbeda dari foto asli, sehingga dapat dideteksi. Penggunaan metode *scale-invariant feature transform* (SIFT) merupakan algoritma yang dapat mendeteksi manipulasi citra "*copy-move forgery*" dan menganalisis citra digital yang dicurigai diedit (Ady Saputro, 2022).

Error level analysis (ELA) adalah salah satu teknik untuk mengetahui suatu gambar asli maupun termodifikasi dengan menghitung perbedaan rata-rata kuantitas nilai Y (*luminance*) dan CrCb (*Chorminance*) untuk menghasilkan konversi warna dominan hitam dan putih pada area gambar asli maupun manipulasi. Secara garis besar ketika gambar digital berformat JPEG pertama kali disimpan, maka itu merupakan gambar asli dengan

kompresi format *original*. (Candra, 2020).

Metode SIFT adalah untuk mengidentifikasi titik-titik kunci atau karakteristik dalam gambar yang telah diubah melalui teknik manipulasi seperti penskalaan, rotasi, dan transformasi *affine*. Tujuannya adalah untuk mendeteksi kejadian pemalsuan *copy-move*. Alasan penggunaan *Scale-Invariant Feature Transform* (SIFT) adalah untuk mendeteksi titik-titik penting dalam suatu gambar dengan menganalisis intensitas ekstrim lokalnya. Titik-titik yang teridentifikasi ini kemudian digunakan untuk membuat deskriptor yang merangkum detail gambar di sekitarnya (Khairan, 2023).

Sehingga cara kerja dari kedua algoritma adalah bahwa ELA digunakan untuk mendeteksi perubahan tingkat kesalahan dalam citra, khususnya yang terkait dengan penyuntingan atau manipulasi, sedangkan SIFT digunakan untuk mengidentifikasi dan mengekstrak fitur-fitur khusus dalam citra yang dapat digunakan untuk keperluan pencocokan atau pengenalan objek.

Jadi penelitian ini untuk menganalisis keaslian suatu foto digital yang telah dimodifikasi atau dimanipulasi menggunakan ELA dan SIFT. Diharapkan dengan penelitian ini dapat meningkatkan pemahaman masyarakat tentang keaslian suatu foto digital, sehingga mereka dapat lebih bijak dalam menilai dan membagikan informasi melalui *platform-platform* digital.

1.2. Rumusan Masalah

Rumusan masalah dalam penelitian ini yaitu sebagai berikut:

1. Bagaimana Implementasi algoritma *Error Level Analysis* (ELA) dan *Scale-Invariant Feature Transform* (SIFT)?
2. Bagaimana hasil analisis akurasi *Error Level Analysis* (ELA) dan *Scale-Invariant*

Feature Transform (SIFT) dalam mendeteksi manipulasi citra digital?

1.3. Batasan Masalah

Batasan masalah dalam penelitian ini yaitu sebagai berikut:

1. Jenis foto yang akan diidentifikasi, adalah foto digital yang diambil oleh kamera digital
2. Sumber foto digital yang akan digunakan dalam penelitian, adalah foto digital yang diambil dari beberapa sumber (relawan), atau yang dihasilkan oleh kamera digital.
3. Metode yang digunakan untuk menentukan keaslian foto digital adalah ELA dan SIFT

1.4. Tujuan Penelitian

Tujuan Penelitian dalam penelitian ini yaitu sebagai berikut:

1. Untuk mengimplementasikan algoritma *Error Level Analysis (ELA)* dan *Scale-Invariant Feature Transform (SIFT)*
2. Untuk menganalisis hasil akurasi *Error Level Analysis (ELA)* dan *Scale-Invariant Feature Transform (SIFT)* sebagai metode deteksi manipulasi citra digital

1.5. Manfaat Penelitian

Manfaat yang di dapat dari penelitian ini yaitu:

1. Penelitian ini dapat berkontribusi pada peningkatan keamanan digital dengan memberikan alat dan teknik yang dapat mengidentifikasi manipulasi foto digital. Ini dapat membantu dalam mencegah penyalahgunaan foto digital yang dapat digunakan dalam tindakan kriminal atau penipuan
2. Dalam konteks ini media sosial dan jurnalisme, penelitian ini dapat membantu dalam mengidentifikasi foto palsu atau manipulasi foto, yang dapat berkontribusi pada upaya pencegahan penyebaran berita palsu.

3. Penelitian ini dapat membantu meningkatkan pemahaman publik tentang potensi manipulasi foto dan bagaimana mengidentifikasinya.

1.6. Sistematis Penulisan

Sistematis penulisan skripsi ini merupakan pembahasan singkat dari setiap bab yang menjelaskan hubungan antara bab satu dengan bab yang lainnya, yaitu sebagai berikut:

BAB I PENDAHULUAN

Terdiri dari latar belakang, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, dan sistematis penulisan.

BAB II TINJAUAN PUSTAKA

Menerapkan teori-teori yang didapat dari sumber-sumber relevan untuk digunakan sebagai panduan dalam penelitian serta menyusun skripsi ini.

BAB III METODE PENELITIAN

Bab ini membahas tentang metode penelitian yang telah dilakukan oleh penulis dengan permasalahan yang diangkat.

BAB IV HASIL DAN PEMBAHASAN

Bab ini membahas tentang hasil dari penelitian yang telah dilakukan yang terdiri dari hasil analisis keaslian foto digital

BAB V PENUTUP

Bab ini memuat kesimpulan dari hasil penelitian yang telah dilakukan, dan saran untuk penelitian selanjutnya mengenai topik terkait

BAB II

TINJAUAN PUSTAKA

2.1. Penelitian Terkait

Dalam penyusunan skripsi ini, penulis terinspirasi dan merujuk pada penelitian-penelitian sebelumnya yang terkait dengan latar belakang masalah pada skripsi ini. Adapun penelitian yang berhubungan dengan skripsi ini dapat dilihat pada tabel 2.1.

Tabel 2.1 Penelitian Terkait

No	Nama dan Tahun	Judul	Hasil Penelitian
1	Made Hari Kesuma Arta, I Gede Bagati Kusuma, Muhammad Rial Rasyid Askar, Ida Bagus Gde Kurnia Dyatmika, Dinda Riani, Lady Hasiani Siahaan (2020)	Analisis Citra Digital Dengan Tools <i>Image Forensic</i> Berupa Foto forensic, <i>Jpegsnoop</i> , Ghiro, Dan <i>Forensically</i>	<i>output</i> yang dihasilkan dari Ghiro, Foto <i>Forensic</i> , dan <i>Forensically</i> terhadap gambar asli dan rekayasa, dapat dilihat Ghiro memberikan gambar yang hampir seluruhnya hitam dengan sedikit noise biru pada bagian bawah gambar. Foto <i>Forensics</i> memberikan gambar hitam yang berisikan noise putih memperlihatkan bentuk gambar asli, tetapi burung pada bagian kiri gambar tidak dapat dilihat secara jelas. <i>Forensically</i> memberikan <i>output</i> gambar hitam dengan <i>noise</i> biru yang membentuk objek serupa dengan gambar asli dan setiap bentuk objek yaitu bison, dan kedua burung dapat dilihat dengan cukup jelas
2	Karolina Eka Purnama, Chaerur Rozikin, Azhari Ali Ridha (2023)	Analisis Forensic Citra Digital Menggunakan Teknik <i>Error Level Analysis</i> dan Metadata Berdasarkan Metode Nist	Hasil penelitian yang telah dilakukan yaitu berupa <i>image forensic</i> untuk membandingkan foto asli dan foto yang sudah dimanipulasi dengan berdasarkan metode

			forensik. Pada penelitian ini dimana menyelesaikan kasus penyebaran foto untuk pemeriksaan foto tersebut dan juga akan melakukan perbandingan pada foto asli dan foto yang telah dimanipulasi
3	Bisri, Hasan Marzuki, Marza Ihsan (2023)	Forensik Citra Digital Menggunakan Metode Error Level Analysis, Clone Detection dan Exif Untuk Deteksi Keaslian Gambar	Hasil dari proses dan penelitian dengan menggunakan laman forensically beta untuk ELA (Error Level Analysis), CD (Clone Detection), Exif (Exchangeable Image File) Metadata dapat di kelompokkan dalam penilaian skoring dengan parameter waktu untuk menghitung lamanya durasi memproses, efektifitas dengan parameter keberhasilan 1 untuk paling rendah dan 10 untuk paling tinggi dalam keberhasilan mengenalisa citra asli.
4	Iis Sudianto, Nuril Anwar (2023)	Analisa Forensik Citra Menggunakan Metode <i>Error Level Analysis</i> Dan <i>Block Matching</i>	Hasil pengujian pada tabel diatas merupakan deteksi manipulasi image menggunakan metode <i>Block Matching</i> . Metode <i>Block Matching</i> menghasilkan image hitam putih yang telah ditandai perbedaan dari kedua image yang sudah diproses menggunakan <i>Error Level Analysis</i> dan <i>Block Matching</i> . Hasil yang didapat berupa direction dua dimensi $f(x,y)$ dimana nilai pixel (x,y) terbesar pada image terdapat bintik putih terbanyak atau banyak perbedaan yang terdapat pada image tersebut

5	Agus Gunawan, Holy Lovenia, Adrian Hartarto Pramudita (2019)	Deteksi Pemalsuan Gambar Dengan Ela Dan <i>Deep Learning</i>	Hasil akurasi yang diperoleh oleh model dalam melakukan klasifikasi dapat dikatakan tergolong tinggi. Hal ini merupakan indikasi bahwa fitur berupa gambar ELA berhasil digunakan untuk melakukan klasifikasi apakah gambar merupakan gambar asli atau sudah mengalami modifikasi.
6	Welson, Deli (2022)	Studi Eksplorasi Perbandingan Fotografi Smartphone Menggunakan Metode Blind Test	Hasil pada penelitian ini mendapatkan rata-rata responden tidak dapat membedakan foto yang diambil dari kamera Android dan <i>iPhone</i> , berdasarkan pada sistem poin <i>blind test</i> yaitu 2189 poin untuk hasil kamera Android dan 2099 poin untuk hasil kamera <i>iPhone</i> . Maka gambar yang dihasilkan kedua <i>smartphone</i> memiliki kualitas yang hampir sama. Pada hasil kuesioner kualitatif mendapatkan kesimpulan bahwa fotografi saat ini sudah banyak dikenali orang-orang terutama pada zaman-zaman sekarang meskipun pengalamannya tidak sedalam dengan fotografer profesional, Baik dalam segi pengetahuan menggunakan jenis-jenis shot, setingan dan Teknik yang digunakan pada shot masing-masing

Berikut ini merupakan beberapa penjelasan untuk membedakan antara penelitian sebelumnya dengan penelitian yang sedang saya teliti. Di antaranya sebagai berikut.

1. Pendekatan Metode:

Judul yang sedang diteliti menggunakan kombinasi metode *Error Level Analysis* (ELA) dan *scale-invariant feature transform* (SIFT). Sebelumnya, terdapat variasi dalam metode yang digunakan, seperti *Deep Learning*, *Block Matching*, dan alat analisis citra digital seperti *Foto Forensic*, *Jpegsnoop*, *Ghiro*, dan *Forensically*.

2. Fokus Penelitian:

Judul sebelumnya mencakup berbagai fokus penelitian, termasuk deteksi pemalsuan gambar, analisis forensik citra digital dengan memanfaatkan metode ELA, dan perbandingan fotografi *smartphone* menggunakan metode *Blind Test*. Judul yang sedang diteliti lebih spesifik dalam identifikasi keaslian foto digital dengan fokus pada deteksi pemalsuan menggunakan ELA dan SIFT.

3. Algoritma dan Teknik:

Judul yang sedang diteliti memadukan ELA dengan *scale-invariant feature transform* (SIFT)", menyoroti penggunaan algoritma dan teknik khusus untuk deteksi manipulasi *copy-move* pada foto digital. Sebaliknya, judul-judul sebelumnya mengintegrasikan berbagai metode seperti *Deep Learning*, *Block Matching*, dan alat analisis citra digital.

4. Referensi Standar dan Metode NIST:

Beberapa judul sebelumnya menyebutkan referensi standar seperti metode NIST dalam analisis forensik citra digital. Judul yang sedang diteliti tidak secara eksplisit menyebutkan metode NIST tetapi berfokus pada ELA dan SIFT.

5. *Tools* dan *Software*:

Judul-judul sebelumnya mencantumkan berbagai alat dan perangkat lunak analisis citra digital seperti *Foto Forensic*, *Jpegsnoop*, *Ghiro*, dan *Forensically*. Judul yang sedang diteliti tidak menyebutkan alat atau perangkat lunak tertentu.

2.2. Analisis

Analisis adalah proses pemecahan atau pemeriksaan yang cermat dan sistematis terhadap suatu informasi, data, atau situasi untuk memahami atau mengevaluasi komponen-komponennya, mengidentifikasi pola, hubungan, atau aspek-aspek penting, serta membuat kesimpulan atau rekomendasi berdasarkan temuan tersebut. Analisis sering digunakan dalam berbagai bidang, seperti ilmu pengetahuan, bisnis, keuangan, teknologi, dan banyak lagi.

Analisis melibatkan langkah-langkah seperti pengumpulan data, pemecahan data menjadi komponen-komponen yang lebih kecil, pengolahan data untuk mengungkap informasi penting, dan pengambilan keputusan berdasarkan temuan tersebut. Tujuan utama dari analisis adalah untuk mendapatkan pemahaman yang lebih baik tentang suatu subjek atau situasi, sehingga bisa digunakan untuk mengambil tindakan yang lebih tepat atau membuat keputusan yang lebih informasi. (Purnama, 2023).

2.3. Foto

Foto adalah kata lain dari kata potret. Foto dalam pengertian secara umum adalah sebuah gambar yang terbentuk dari sebuah jepretan kamera. Foto atau fotografi berasal dari bahasa Inggris yaitu *Photography*, sedangkan *Photography* itu sendiri dari bahasa Yunani yang terdiri dari kata *photos* yang memiliki arti cahaya dan *grafo* memiliki arti menulis atau melukis menggunakan bantuan media cahaya. Selain definisi yang dijelaskan diatas, foto juga dapat dibedakan menjadi bermacam-macam kategori. Maksud dari pembagian kategori ini sendiri bertujuan agar lebih mudah dalam pembuatan dan pemanfaatannya, tinggal menyesuaikan keperluan dari masing-masing individu itu sendiri. Ada cukup banyak kategori dari foto itu sendiri, diantaranya foto jurnalistik, foto kedokteran, foto olahraga, dan

masih banyak lagi (Novianti, 2023).

2.4. Citra

Citra merupakan sebuah gambar yang merepresentasikan suatu objek. Citra digital merupakan representasi objek yang dapat diolah di dalam perangkat komputer. Citra digital diperoleh dari peralatan digital seperti kamera digital sehingga dapat secara langsung diolah di dalam komputer. Penyebab citra yang dihasilkan dari peralatan digital dapat secara langsung diolah di dalam komputer adalah terdapat sistem sampling dan kuantisasi yang berguna untuk mengubah citra kontinu menjadi digital serta mengubah *aperture*, *white balance*, *metering*, tanggal dan jam pemotretan, dan *focal length* lensa yang mana data ini tidak bisa didapatkan jika menggunakan kamera *analog* (Bisri, 2023).

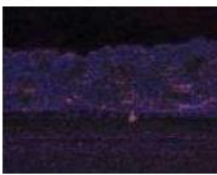
Citra asli adalah citra yang didapatkan dari hasil peralatan digital seperti kamera digital yang diambil secara langsung yang merujuk pada gambar atau foto yang belum mengalami perubahan atau manipulas (Purnama, 2023).

Manipulasi Citra Manipulasi citra merupakan proses pengolahan citra dengan mengubah citra yang asli menjadi citra lain yang diinginkan. Proses ini dapat bertujuan untuk memperbaiki atau memperbagus sebuah citra yang kurang baik (mengandung *noise*, *blur*, dan lainnya), ataupun memberikan informasi yang salah (Suhailie, 2021).

2.5. Analisis Citra

Analisis citra adalah kegiatan menganalisis citra sehingga menghasilkan informasi untuk menetapkan keputusan. Beberapa hal yang dapat diidentifikasi dari sebuah citra seperti format file citra, ukuran file citra, jumlah *pixel*, dimensi citra, resolusi citra, dan lain-lain. Ada berbagai macam perangkat lunak yang dapat kita gunakan untuk memperoleh informasi dari sebuah citra seperti *Microsoft Paint*, *Microsoft Office Picture Manager*, *Adobe*

Berikut merupakan contoh cara kerja dari Metode ELA dapat dilihat pada gambar 2.1.

Nama Image	Kompresi ELA 95%)	MSE dan PSNR	Keterangan
 Nama Image :beach_1		MSE : 24.6574 PSNR : 34.2453	Hasil nilai MSE dari kedua image menunjukkan bahwa image 2 memiliki nilai MSE yang lebih besar dan nilai PSNR lebih kecil dibandingkan image 1, sehingga disimpulkan bahwa image 2 merupakan image yang telah dilakukan editing
 Nama Image :beach_2		MSE : 37.5131 PSNR : 32.423	

Gambar 2.1 Contoh cara kerja ELA (Sudianto, 2023).

2.7. Scale-Invariant Feature Transform (SIFT)

Scale-Invariant Feature Transform (SIFT) adalah sebuah teknik atau metode deteksi manipulasi citra yang dikenal sebagai "*copy-move forgery*" (pemalsuan dengan menyalin dan memindahkan bagian dari citra ke bagian lainnya). Teknik ini menggunakan algoritma *Scale-Invariant Feature Transform (SIFT)* sebagai dasarnya.

Copy-move forgery adalah tindakan memindahkan atau menyalin satu atau beberapa bagian dari citra digital dan menempatkannya dalam area lain dalam citra yang sama. Tujuan dari tindakan ini bisa beragam, seperti menghilangkan objek yang tidak diinginkan dari citra atau mengganti area tertentu untuk memanipulasi konten citra (Singh 2022).

Teknik SIFT bekerja dengan cara mengekstrak fitur-fitur unik dari citra, yang disebut dengan "*keypoints*," menggunakan algoritma SIFT. Kemudian, algoritma ini mencari kesamaan atau kesamaan antara beberapa keypoints dalam citra yang mungkin mengindikasikan adanya pemalsuan *copy-move*. Jika kesamaan ditemukan, itu bisa menandakan bahwa ada bagian dari citra yang telah disalin dan dipindahkan ke area lain

dalam citra.

Maka dengan ini menggunakan algoritma SIFT untuk mengidentifikasi titik-titik kunci atau karakteristik dalam gambar yang telah diubah melalui teknik manipulasi seperti penskalaan, rotasi, dan transformasi *affine*. Tujuannya adalah untuk mendeteksi kejadian pemalsuan *copy-move*. Alasan penggunaan *Scale-Invariant Feature Transform* (SIFT) adalah untuk mendeteksi titik-titik penting dalam suatu gambar dengan menganalisis intensitas ekstrim lokalnya. Titik-titik yang teridentifikasi ini kemudian digunakan untuk membuat deskriptor yang merangkum detail gambar di sekitarnya. Karakteristik ini selanjutnya dapat digunakan dalam berbagai tugas seperti mencocokkan gambar, mengenali objek, dan mengambil gambar (Salin, 2023). Contoh cara kerja metode SIFT dapat dilihat pada gambar 2.2.



Figure 2. Original Evidence.



Gambar 2.2 Contoh cara kerja metode SIFT (Khairan, 2023).

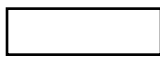
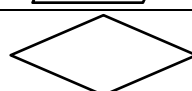
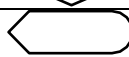
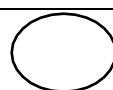
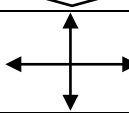
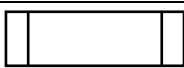
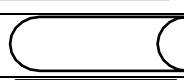
2.8. Flowchart

Flowchart merupakan penyajian yang sistematis tentang proses dan logika dari kegiatan penanganan informasi atau penggambaran secara grafik dari Langkah-langkah dan urutan prosedur dari suatu program. Bagan alir (*flowchart*) adalah bagan (*chart*) yang menunjukkan alir (*flow*) di dalam program atau prosedur sistem secara logika. Bagan

alir digunakan terutama untuk alat bantu komunikasi dan untuk dokumentasi (Rejeki 2013).

Menurut Sitorus (2015), *Flowchart* merupakan urutan logika dari suatu prosedur pemecahan masalah, sehingga *flowchart* merupakan langkah-langkah penyelesaian masalah yang dituliskan dalam simbol-simbol tertentu. Diagram alir ini akan menunjukkan alur didalam program secara logika. *Flowchart* bertujuan untuk menggambarkan suatu tahapan penyelesaian masalah secara sederhana, terurai rapi dan jelas menggunakan simbol-simbol standa (Charis Noiija, 2023). Berikut dapat dilihat pada tabel 2.3.

Tabel 2.3 Simbol dan Fungsi *Flowchart* (Charis, 2023).

No	Simbol	Fungsi
1		Terminal, untuk memulai dan mengakhiri suatu proses/ kegiatan.
2		Proses, suatu yang menunjukan setiap pengolahan yang dilakukan oleh komputer.
3		Adalah step khusus deklarasi variable dengan <i>flowchart</i>
4		Input, untuk memasukan hasil dari suatu proses.
5		<i>Decision</i> , suatu kondisi yang akan menghasilkan beberapa kemungkinan jawaban atau pilihan
6		<i>Display, output</i> yang ditampilkan dilayar terminal
7		<i>Connector</i> , suatu prosedur akan masuk atau keluar melalui simbol ini dalam lembar yang sama.
8		<i>Off Page Connector</i> , merupakan simbol masuk atau keluarnya suatu prosedur pada kertas lembar lain.
9		<i>Arus Flow</i> , symbol ini digunakan untuk menggambarkan arus proses dari,suatu,kegiatan lain.
10		<i>Hard Disk Storage, input output</i> yang menggunakan <i>hard disk</i> .
11		<i>Predifined Process</i> , untuk menyatakan sekumpulan langkah proses yang ditulis sebagai prosedur
12		<i>Stored Data, input, output</i> yang menggunakan disket.
13		Printer, simbol ini digunakan untuk menggambarkan suatu dokumen atau kegiatan untuk mencetak suatu informasi dengan mesin printer.

2.9. Python

Python merupakan bahasa pemrograman interpretatif yang berfokus pada keterbacaan kode. Python diklaim sebagai bahasa pemrograman yang memiliki kode-kode pemrograman yang jelas, lengkap, dan mudah untuk dipahami. Python secara umum berbentuk pemrograman berorientasi objek, pemrograman imperatif, dan pemrograman fungsional. Python dapat digunakan dalam berbagai keperluan pengembangan perangkat lunak dan dapat berjalan di berbagai platform sistem operasi seperti *Linux*, *Windows*, *Mac OS X*, dan *Symbian*.

Bahasa pemrograman Python pertama kali diperkenalkan pada tahun 1991 oleh *Guido van Rossum* sebagai *web development software development*, *mathematics*, dan *system scripting*. Python dapat digunakan pada server untuk membuat aplikasi berbasis web, dapat dihubungkan dengan sistem basis data, dapat membaca dan memodifikasi file, menangani *big data* dan aplikasi matematika yang kompleks (Bisri, 2023).

2.10. Library

Library, atau sering disebut juga pustaka, adalah kumpulan modul atau fungsi yang siap pakai untuk digunakan dalam pengembangan perangkat lunak. Fungsinya adalah mempercepat proses pengembangan dengan menyediakan implementasi fungsi umum, algoritma, atau fitur tertentu. Pengembang dapat menghemat waktu dan upaya karena mereka dapat menggunakan fungsionalitas yang sudah ada, tanpa perlu menulis ulang kode dari awal. Selain itu, *library* memungkinkan pengembang untuk memanfaatkan solusi yang telah diuji dan diverifikasi oleh komunitas pengembang sebelumnya (Prabowo, 2022).

2.11. White box

White box testing disebut sebagai pengujian *structural*. Yang mana perangkat lunak

yang diuji merupakan hal transparan kepada penguji. Dalam pengujian, uji dirancang dari perspektif pengembang dikarenakan struktur internal dikenal dengan menguji segala bagian kode yang mampu untuk diuji dengan tujuan untuk menentukan kesalahan logis dari kode sumber perangkat lunak (Praniffa, 2023).

Teknik basis *path* merupakan salah satu teknik pengujian *white box*. Teknik ini memungkinkan penguji untuk mengukur kompleksitas logika dari rancangan procedural. Skenario uji coba yang dibuat untuk menguji teknik basis *path* ini dijamin akan mengeksekusi setiap *statement* dari aplikasi yang diujikan setidaknya satu saat tahap pengujian.

Cyclomatic complexity adalah besaran perangkat lunak yang menyediakan acuan kuantitatif kompleksitas suatu logika dalam program. Rumus menghitung *Cyclomatic complexity* adalah sebagai berikut:

$$V(G) = E - N + 2 \dots\dots\dots(2.4)$$

Keterangan 2.4:

E =Jumlah *edges* pada *flowgraph*.

N =Jumlah *nodes* pada *flowgraph*.

P =Jumlah *predicates nodes* pada *flowgraph*.

2.12. Agile

Agile adalah suatu pendekatan atau metodologi pengembangan perangkat lunak yang menekankan pada kolaborasi tim, adaptasi terhadap perubahan, pengiriman perangkat lunak yang berfungsi, dan respons cepat terhadap umpan balik. Pendekatan ini berfokus pada pengembangan iteratif dan inkremental, yang memungkinkan tim untuk merancang, menguji, dan mengirimkan potongan-potongan kecil dari perangkat lunak secara berkala. Metode *agile* merupakan salah satu metodologi di tahapan pengembangan

sistem yang sudah banyak digunakan (Nova, 2022).

Adapun beberapa model di dalam metode *agile*, sebagai berikut:

1. *Scrum*

Model *scrum* merupakan model yang mengedepankan kecepatan pengembangan. Didalam prosesnya, rencana proyek sistem informasi diperiksa terus menerus dan disesuaikan dengan kenyataan pada proyek tersebut.

2. *Agile*

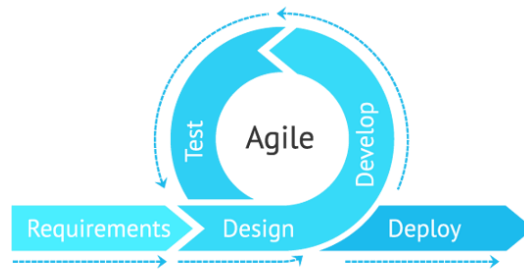
Model *Agile* merupakan pendekatan pengembangan sistem yang memungkinkan perubahan setiap saat. Pendekatan ini berfokus pada pengembangan iteratif dan inkremental, yang memungkinkan tim untuk merancang, menguji, dan mengirimkan potongan-potongan kecil dari perangkat lunak secara berkala. Metode *agile* merupakan salah satu metodologi di tahapan pengembangan sistem yang sudah banyak digunakan.

3. *Extreme Programming (XP)*

Model *Extreme Programming* merupakan pengembangan perangkat lunak seperti sistem informasi yang mempunyai sasaran pembangunan dengan menetapkan kebutuhan yang belum jelas ataupun perubahan terhadap kebutuhan yang sangat cepat, serta dengan tim yang kecil sampai menengah.

4. *Spiral*

Model *spiral* merupakan model yang menekankan lebih banyak analisis risiko. Model ini mempunyai empat *fase*, yaitu *planning* atau perencanaan, *risk analysis* atau analisis risiko, *engineering* atau implementasi, dan *evaluation* atau evaluasi dari pembuatan sistem informasi tersebut. Sehingga dapat memudahkan dalam merancang suatu *system*. Berikut dapat dilihat pada gambar 2.3.



Gambar 2.3 Metode *Agile* (Nova, 2022).

2.13. Model *Agile*

Metode rekayasa perangkat lunak yang digunakan dalam perancangan sistem ini adalah Pendekatan *Agile Development*. Dalam pendekatan *agile*, pengembangan perangkat lunak dimungkinkan untuk dilakukan dalam waktu yang singkat (1-3 bulan) dengan pengembangan fitur satu per satu. Peneliti akan mencari pemecahan masalah yang diselidiki dengan merancang suatu sistem berdasarkan kebutuhan dari objek penulisan. Pelaksanaan metode rekayasa meliputi seluruh aspek produksi perangkat lunak, mulai dari tahap analisa kebutuhan pengguna, perancangan komponen-komponen sistem, pengkodean, pengujian sistem, *release*, revisi dan evaluasi sampai pada tahap pemeliharaan sistem (Pratasik, 2020).

2.14. Metode Penelitian

Metode penelitian adalah prosedur dan skema yang digunakan dalam penelitian. Metode penelitian memungkinkan penelitian dilakukan secara terencana, ilmiah, netral dan bernilai. Metode penelitian sebagai strategi mengumpulkan data, dan menemukan solusi suatu masalah berdasarkan fakta. (Waruwu, 2023).

Secara umum ada tiga metode penelitian yang umum digunakan dalam penelitian ilmiah. Ketiga metode penelitian tersebut terdiri dari metode penelitian kuantitatif, metode penelitian kualitatif, dan metode penelitian dan pengembang.

2.14.1. Penelitian Kualitatif

Metode penelitian kualitatif adalah metode penelitian yang berlandaskan pada filsafat postpositivisme, digunakan untuk meneliti pada kondisi obyek yang alamiah, (sebagai lawannya adalah eksperimen) dimana peneliti adalah sebagai instrumen kunci, teknik pengumpulan data dilakukan secara triangulasi (gabungan), analisis data bersifat induktif/kualitatif, dan hasil penelitian kualitatif lebih menekankan makna dari pada generalisasi (Waruwu, 2023).

2.14.2. Penelitian Kuantitatif

Metode kuantitatif dinamakan metode tradisional, karena metode ini sudah cukup lama digunakan sehingga sudah mentradisi sebagai metode untuk penelitian. Metode ini disebut sebagai metode positivistik karena berlandaskan pada filsafat positivisme. Metode ini sebagai metode *ilmiah/scientific* karena telah memenuhi kaidah-kaidah ilmiah yaitu konkrit, empiris, obyektif, terukur, rasional, dan sistematis. Metode ini juga disebut metode discovery, karena dengan metode ini dapat ditemukan dan dikembangkan berbagai iptek baru. Metode ini disebut metode kuantitatif karena data penelitian berupa angka-angka dan analisis menggunakan statistic (Waruwu, 2023).

2.14.3. Penelitian dan Pengembangan

Metode penelitian dan pengembangan atau dalam bahasa Inggrisnya *Research and Development* adalah metode penelitian yang digunakan untuk menghasilkan produk tertentu, dan menguji keefektifan produk tersebut. Untuk dapat menghasilkan produk tertentu digunakan penelitian yang bersifat analisis kebutuhan dan untuk menguji keefektifan produk tersebut supaya dapat berfungsi di masyarakat luas, maka diperlukan penelitian untuk menguji keefektifan produk tersebut. Jadi penelitian dan pengembangan bersifat

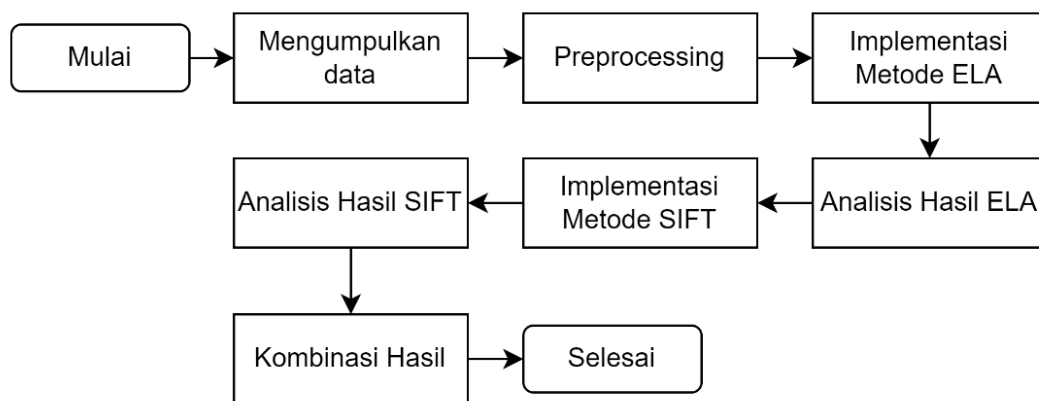
longitudinal (bertahap bisa *multy years*). Penelitian Hibah Bersaing (didanai oleh Direktorat Jenderal Pendidikan Tinggi), adalah penelitian yang menghasilkan produk, sehingga metode yang digunakan adalah metode penelitian dan pengembangan (Waruwu, 2023).

BAB III

METODE PENELITIAN

3.1. Alur Penelitian

Berikut ini merupakan alur dari penelitian yang akan di lakukan. Dapat dilihat pada gambar 3.1.



Gambar 3.1 Alur Penelitian.

3.2. Objek dan Waktu Penelitian

Objek yang diteliti dalam penelitian ini adalah citra digital atau foto yang menjadi objek analisis. Hasil foto yang diambil oleh kamera digital, serta hasil edit atau manipulasi citra, merupakan fokus utama penelitian ini. Rentang waktu penelitian harus mencakup periode ketika data sampel citra digital atau foto yang akan dianalisis diperoleh. Pada periode ini, analisis dilakukan menggunakan metode *Error Level Analysis* (ELA) dan *Scale-Invariant Feature Transform* (SIFT).

Penentuan waktu penelitian juga melibatkan fase-fase penting, seperti waktu pengumpulan sampel citra, pengolahan data, dan analisis. Penting untuk mencatat bahwa kalimat baku tidak boleh diawali dengan kata penghubung. Penelitian ini dilakukan pada semester ganjil tahun ajaran 2023/2024.

3.3. Metode Pengumpulan Data

Metode pengumpulan data yang digunakan dalam penelitian ini menggunakan metode observasi terhadap objek yang diteliti yaitu terhadap citra digital secara langsung dan sampel yang digunakan pada penelitian ini bersumber dari beberapa rekan-rekan yang bersedia memberikan sampel citra.

Berikut merupakan pengumpulan data dari penelitian yang sedang dikembangkan:

1. Data yang dikumpulkan berupa sampel foto yang diambil dengan kamera digital. Citra-citra ini harus mencakup berbagai jenis foto dan menggambarkan berbagai situasi dan kondisi pencahayaan.
2. Data yang dikumpulkan juga berupa sampel hasil edit yang telah dimanipulasi oleh *tools*. Citra ini harus mencakup berbagai jenis manipulasi, seperti retus, perubahan warna, atau pemindahan objek.
3. *Tools* yang digunakan adalah perangkat lunak untuk mendukung analisis *Error Level Analysis* (ELA) untuk menghasilkan data tingkat kesalahan (*error level*) pada citra foto dan citra hasil manipulasi. Peneliti melakukan analisis ELA pada semua sampel citra potret dan citra hasil edit yang telah Anda kumpulkan. Hasil analisis ELA akan menghasilkan informasi tentang tingkat kesalahan pada citra tersebut.
4. Kemudian peneliti mengimplementasikan metode *Scale-Invariant Feature Transform* (SIFT) pada sampel citra potret dan citra hasil edit yang telah dikumpulkan.
5. Peneliti melakukan analisis untuk mendeteksi tanda-tanda pemalsuan dengan menyalin dan memindahkan bagian dari citra ke area lain dalam citra.
6. Simpan data hasil analisis dari metode *Error Level Analysis* (ELA) dan *Scale-Invariant Feature Transform* (SIFT) untuk setiap sampel citra

3.4. Alat dan Bahan Penelitian

Dalam melakukan penelitian ini, ada beberapa spesifikasi alat penelitian yang harus dipenuhi. Spesifikasi alat penelitian maksudnya adalah standar minimal dari alat (*tools*) yang digunakan sebagai wadah utama dalam melakukan perbandingan penelitian ini untuk mengidentifikasi keaslian foto digital dan foto manipulasi. Alat yang digunakan antara lain sebagai berikut:

3.4.1. Perangkat keras (*Hardware*)

Adapun spesifikasi perangkat keras (*hardware*) yang digunakan pada penelitian ini sebagai penunjang proses perancangan dan pembuatan sistem selama penelitian berlangsung. Spesifikasi *hardware* dapat dilihat pada tabel 3.1.

Tabel 3.1 Detail Spesifikasi *Hardware*.

No	Jenis	Spesifikasi
1.	Leptop	HP Spectre x360 13
2.	<i>Processor</i>	Intel Core i5 gen 8
3.	Memori (RAM)	4 GB
4.	SSD	512 GB
5.	<i>System Type</i>	64-bit
6	<i>Android</i>	12, XOS 10.6

3.4.2. Perangkat Lunak (*software*)

Selain kebutuhan *hardware* yang memadai, penulis juga membutuhkan *software* yang mendukung untuk melakukan perancangan dan pembuatan sistem, guna mengidentifikasi keaslian foto digital dan foto manipulasi. Spesifikasi *software* dapat dilihat pada tabel 3.2.

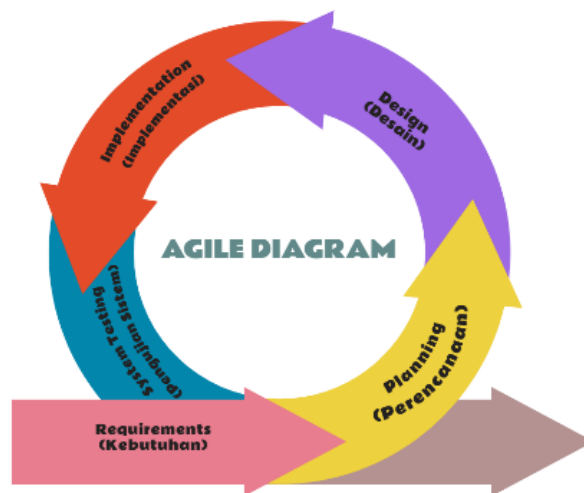
Tabel 3.2 Spesifikasi *Software*.

No	Jenis	Spesifikasi	Keterangan
1.	Sistem Operassi	<i>Windows 11</i>	Sistem Operasi yang digunakan selama penelitian
2.	Aplikasi Teks Editor	Python Versi 3.7.9, <i>jupyter Notebook & Visual Studio Code, rapidminer, android studio</i>	Digunakan untuk menulis dan menjalankan <i>script</i> python

3	Vscode	Visual Studio Code	Visual Studio Code (VSCode) adalah sebuah editor kode yang digunakan peneliti
4	Android	12, XOS 10.6	Sistem operasi <i>mobile</i> yang digunakan untuk menampilkan hasil peneliti.

3.5. Metode Pengembang Sistem

Metode pengembangan sistem menggunakan *Agile*, adapun langka-langka *agile* pada penelitian ini dapa dilihat pada gambar 3.2.



Gambar 3.2 Model Agile.

Berikut merupakan tahap-tahap model *agile* yaitu:

3.5.1. *Requirements* (kebutuhan)

1. Mengelompokkan Data:

Foto-foto yang akan dianalisis perlu dikelompokkan. Mungkin melibatkan pengumpulan dataset foto digital dari berbagai sumber, termasuk foto asli dan foto yang telah mengalami manipulasi.

2. *Pre-prcessing*:

Pre-processing dalam konteks *Error Level Analysis* (ELA) dan *Scale-Invariant Feature Transform* (SIFT) melibatkan serangkaian langkah untuk mempersiapkan citra

digital sebelum dilakukan analisis lebih lanjut. Proses ini bertujuan untuk memastikan bahwa citra yang digunakan dalam analisis memiliki karakteristik yang sesuai dengan metode analisis yang akan diterapkan, baik itu untuk mendeteksi kesalahan level atau mengekstrak fitur skala invarian. *Pre-processing* dapat mencakup konversi format, normalisasi warna, pengurangan *noise*, dan langkah-langkah lain yang diperlukan untuk meningkatkan keakuratan dan relevansi hasil analisis.

Tahapan *Pre-processing*:

a. Format

Memastikan semua foto dalam format yang sesuai untuk analisis. Jika perlu, mengonversi format foto ke format yang lebih standar.

b. Normalisasi Warna

Melakukan normalisasi warna pada foto untuk mengatasi perbedaan dalam pencahayaan atau warna. Tujuannya adalah membuat foto seragam sehingga analisis lebih konsisten.

c. Reduksi *Noise*

Mengurangi *noise* atau gangguan pada foto yang dapat mempengaruhi hasil analisis. Ini bisa melibatkan penggunaan filter atau teknik lainnya untuk membersihkan data dari gangguan.

d. Normalisasi *Histogram*

Menyesuaikan *histogram* foto untuk memperoleh distribusi intensitas piksel yang sesuai. Ini membantu meningkatkan kualitas dan kemudahan analisis.

e. Segmentasi

Melakukan segmentasi untuk memisahkan foto menjadi bagian-bagian yang relevan

atau untuk mengidentifikasi area tertentu yang akan dianalisis lebih lanjut.

f. *Resize*

Menyesuaikan ukuran foto jika diperlukan agar sesuai dengan persyaratan analisis atau untuk mengurangi kompleksitas komputasi.

3. Analisis Menggunakan ELA

Foto asli dianalisis menggunakan *Error Level Analysis* (ELA). Proses ELA juga diterapkan pada foto yang telah diedit oleh *tools*. ELA membantu mengidentifikasi perbedaan tingkat kesalahan piksel, memungkinkan deteksi potensial manipulasi.

4. Analisis Menggunakan *Scale-Invariant Feature Transform* (SIFT):

Diagram alur dapat mencakup cabang kedua untuk analisis foto menggunakan *Scale-Invariant Feature Transform* (SIFT). Metode ini digunakan untuk mendeteksi apakah ada elemen foto yang disalin dan dipindahkan dalam foto.

5. Pembuatan Model

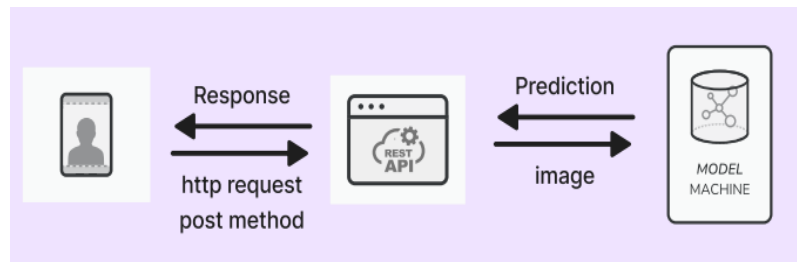
Kemudian diperlukan pembuatan model untuk meningkatkan akurasi deteksi. Pilih model *machine learning* yang sesuai, latih dengan dataset yang sesuai, dan evaluasi kinerja model.

6. Pembuatan *Application Programming Interface* (API)

Pembuatan antarmuka merujuk pada proses pengembangan dan penentuan standar yang memungkinkan aplikasi perangkat lunak untuk berkomunikasi satu sama lain.

3.5.2. **Planning (Perencanaan)**

Pada tahap perencanaan, peneliti akan merancang serangkaian arsitektur pengguna berdasarkan kebutuhan yang telah ditentukan sebelumnya. Dapat dilihat pada gambar 3.3



Gambar 3.3 Perancangan arsitektur.

Penjelasan:

- Request POST* method digunakan saat ingin mengirimkan data *input*, *form submissions*, ataupun upload file ke *server*.
- User mengirimkan *request POST* berisi data *input image* ke *Rest API*
- Rest API* menerima *request POST* dengan data image dari *user*
- Rest API* kemudian meneruskan data image tersebut ke *Prediction Model* (model prediksi)
- Prediction model* selanjutnya memproses data *image* yang diterima dari *Rest API*
- Prediction model* melakukan klasifikasi/prediksi berdasarkan data *image* yang dikirim
- Hasil prediksi dari model kemudian dikirimkan kembali ke *Rest API*
- Rest API* menerima hasil prediksi dari model lalu mengirimkannya ke *user*
- User menerima hasil prediksi melalui *response* dari *request POST* nya ke *Rest API*

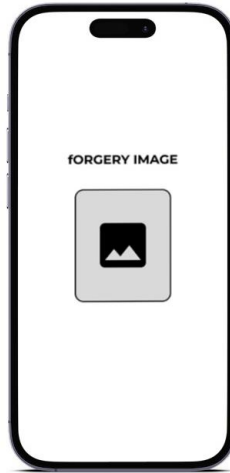
3.5.3. Design (Desain)

Pada tahap ini, peneliti merancang desain aplikasi dan melakukan pemodelan untuk mengidentifikasi keaslian foto digital. Desain ini hanya dimaksudkan untuk memenuhi kebutuhan dari pengguna yang sedang dikerjakan. Berikut adalah desain tampilan aplikasi:

- Tampilan Awal

Tampilan awal adalah halaman yang muncul ketika sebuah aplikasi pertama kali dibuka. Halaman ini akan tampil selama beberapa detik sebelum tampilan utama

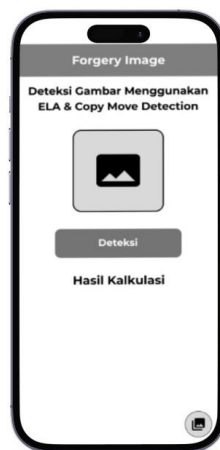
aplikasi diperlihatkan kepada pengguna. Dapat dilihat pada gambar 3.4.



Gambar 3.4 Tampilan Awal Aplikasi.

b. Tampilan Utama

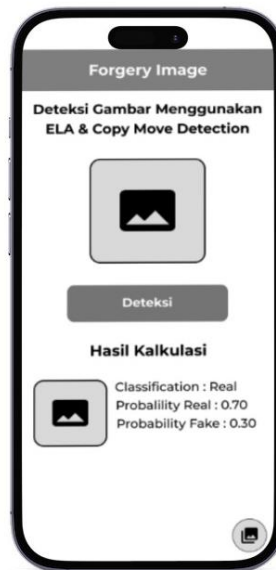
Tampilan Utama merupakan halaman setelah tampilan awal, atau dapat disebut sebagai halaman utama, dimana terdapat app bar yang berisi ikon menu untuk mendeteksi foto. Dapat dilihat pada gambar 3.5.



Gambar 3.5 Tampilan Utama Aplikasi.

c. Tampilan hasil foto asli

Tampilan hasil foto asli yang telah di analisis keakurasianya. Dapat dilihat pada gambar 3.6.



Gambar 3.6 Tampilan hasil foto asli.

d. Tampilan hasil foto manipulasi

Dalam tampilan ini, foto yang telah dideteksi akan menampilkan hasil foto yang telah dimanipulasi. Dapat dilihat pada gambar 3.7.



Gambar 3.7 Tampilan hasil manipulasi

3.5.4. *Implementation* (Implementasi)

Pada tahap ini, peneliti melakukan pembuatan kode program berdasarkan desain yang telah dirancang sebelumnya. Setelah itu, kode program diuji. Tahap ini terdiri dari tiga

sub-tahap, yaitu unit *testing*, penulisan kode, dan perbaikan kode (*refactor*). Implementasi dianggap selesai jika kode berjalan tanpa kesalahan (*error*) dan pengujian unit berhasil.

3.5.5 System Testing (Pengujian Sistem)

Ini adalah tahap pengujian perangkat lunak dimana keseluruhan sistem yang terintegrasi diuji menggunakan *white box*. Tujuannya adalah memastikan bahwa hasil implementasi sesuai dengan kebutuhan yang telah ditentukan sebelumnya. Setiap kekurangan atau cacat yang ditemukan dicatat dan diperbaiki.

3.6. Impementasi Library

Berikut ini merupakan impementasi *library* dapat dilihat pada tabel 3.3.

Tabel 3.3 Penerapan *Library*.

No	Library	Penjelasan
1	<code>Import warnings dan warnings.filterwarnings("ignore")</code>	Ini digunakan untuk mengabaikan peringatan yang mungkin dihasilkan selama eksekusi program. Dengan mengatur filter peringatan menjadi "ignore", program akan mengabaikan peringatan dan tidak menampilkannya.
2	<code>Import matplotlib.pyplot as plt:</code>	Modul ini digunakan untuk membuat visualisasi grafik dan plot. Biasanya digunakan untuk menampilkan gambar atau grafik hasil eksperimen atau data.
3	<code>import numpy as np:</code>	<i>NumPy</i> adalah <i>library</i> yang menyediakan dukungan untuk <i>array</i> dan operasi matematika pada <i>array</i> . Sering digunakan untuk manipulasi data numerik
4	<code>import pandas as pd:</code>	<i>Pandas</i> adalah <i>library</i> untuk analisis data yang menyediakan struktur data fleksibel bernama <i>DataFrame</i> , yang sangat berguna untuk bekerja dengan data tabular.
5	<code>import os:</code>	Modul ini menyediakan fungsi-fungsi untuk berinteraksi dengan sistem operasi, seperti mengakses file dan direktori.
6	<code>from PIL import Image:</code>	<i>Python Imaging Library</i> (PIL) digunakan untuk memanipulasi dan bekerja dengan gambar.
7	<code>import matplotlib:</code>	Ini adalah <i>library</i> visualisasi data yang

		mencakup fungsi-fungsi umum untuk pembuatan grafik dan plot.
8	<code>from sklearn.model_selection import train_test_split:</code>	Digunakan untuk membagi data menjadi set pelatihan dan pengujian.
9	<code>import tensorflow as tf:</code>	<i>TensorFlow</i> adalah <i>library opensource</i> yang sering digunakan untuk pembuatan dan pelatihan model <i>machine learning</i> .
10	<code>import keras:</code>	Keras adalah <i>high-level neural networks</i> API yang berjalan di atas <i>TensorFlow</i> . Keras menyediakan antarmuka yang lebih sederhana untuk membangun dan melatih model <i>neural networks</i> .
11	<code>from keras.models import *, from keras.layers import *, from keras.callbacks import *, from keras.optimizers import</code>	Mengimpor beberapa kelas dan fungsi yang diperlukan dari Keras untuk membangun, melatih, dan mengevaluasi model <i>neural networks</i>
12	<code>from tqdm import tqdm:</code>	Menyediakan bar progres saat iterasi atau <i>loop</i>
13	<code>import cv2:</code>	<i>OpenCV</i> adalah <i>library</i> komputer vision yang sering digunakan untuk memproses dan menganalisis gambar.
14	<code>import datetime:</code>	Digunakan untuk bekerja dengan data waktu dan tanggal.

3.7. Contoh Kasus

Pada penelitian ini contoh kasus yang akan dibuat yaitu untuk menganalisis sebuah foto yang telah mengalami pengeditan atau yang telah dimanipulas.

Kasus manipulasi citra yang terjadi saat ini, misalnya, adalah beredarnya foto-foto palsu yang sensitif sehingga menimbulkan informasi palsu kepada masyarakat, sehingga menimbulkan permasalahan-permasalahan dalam kehidupan sosial, seperti penyebaran informasi-informasi yang tidak benar, hingga sangat mudah terjadi kesalah pahaman, seperti penipuan *online*.

3.8. Perencanaan Pengujian Sistem

Pada penelitian ini metode pengujian yang digunakan adalah *White box*. Bagian yang akan diuji adalah uji data prediksi. Adapun Langkah-langkah pengujian sebagai berikut:

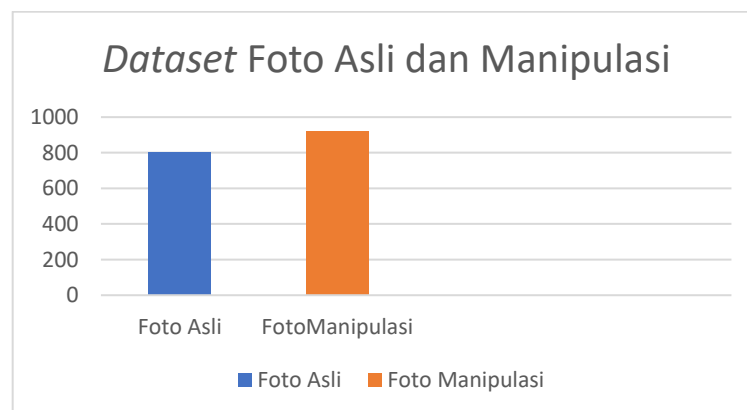
1. Membuat *flowchart* prediksi
2. Membuat *flowgraph* prediksi
3. Mencari *cyclomatic complexity*
4. Tentukan *independent part*

BAB IV

HASIL DAN PEMBAHASAN

4.1. Analisis Data

Pada tahapan ini menjelaskan tentang pembuatan model sekaligus melakukan perancangan sebuah sistem yang bertujuan untuk Analisis Keaslian Foto Digital Menggunakan *Error Level Analysis* (ELA) dan *Scale-Invariant Feature Transform* (SIFT). Dataset yang diperoleh dikaggle yang sudah divalidasi. *Dataset* yang dikumpulkan berdasarkan jenis-jenis foto yaitu foto asli sebanyak 800 foto dan foto manipulasi sebanyak 921 foto. Jadi *dataset* keseluruhan terdapat 1721 foto yang nantinya akan digunakan untuk *dataset*. Untuk lebih jelas dapat dilihat pada gambar 4.1.



Gambar 4.1 *Dataset* Foto Asli dan Manipulasi.

4.2. Implementasi ELA dan SIFT

Berdasarkan dataset yang telah diolah menggunakan *Error Level Analysis* (ELA) dan *Scale-Invariant Feature Transform* (SIFT) sebelumnya kemudian dilanjutkan pada proses pengimplementasian pada data latih yang sudah dibagi pada tahapan sebelumnya. Berikut penjelasan terkait bagaimana pengimplementasian disini menjelaskan penerapan pada ELA berdasarkan foto-foto dataset yang sudah diolah menggunakan ELA kemudian

diimplementasikan pada:

1. *Import Library*

Program ini menggunakan beberapa perpustakaan diantaranya adalah PIL (*Pillow*) yaitu untuk manipulasi gambar. *Matplotlib* untuk visualisasi gambar. Sedangkan *cv2* (*OpenCV*) yaitu untuk pemrosesan gambar, termasuk kompresi dan perhitungan perbedaan. Untuk lebih jelas dapat dilihat pada gambar 4.2.



```
from PIL import Image, ImageFilter, ImageChops
import matplotlib.pyplot as plt
import cv2
```

Gambar 4.2 *Import Library*.

2. *Konfigurasi Kelas (Class Config)*

Class ini mendefinisikan konfigurasi untuk model *machine learning*. Namun, dalam konteks program ini, *class* ini tidak digunakan lebih lanjut. Untuk lebih jelas dapat dilihat pada gambar 4.3.



```
class Config:
    epochs = 30
    batch_size = 32
    lr = 1e-3
    name = 'xception'
    n_labels = 2
    image_size = (224, 224)
    decay = 1e-6
    momentum = 0.95
    nesterov = False
```

Gambar 4.3 *Konfigurasi Kelas (Class Config)*.

3. *Fungsi compute ELA Cv*

Fungsi ini melakukan operasi ELA dengan langkah-langkah berikut yaitu Membaca gambar asli dari path yang diberikan, mengonversinya dari BGR ke RGB, menyimpannya

ulang dengan tingkat kompresi JPEG yang ditentukan, membaca kembali gambar yang telah dikompres, menghitung perbedaan mutlak antara gambar asli dan yang dikompres, lalu mengalikan hasil perbedaan dengan skala tertentu untuk memperjelas perbedaan. Untuk lebih jelas dapat dilihat pada gambar 4.4.

```
def compute_ela_cv(path, quality):
    temp_filename = 'temp_file_name.jpg'
    SCALE = 15
    orig_img = cv2.imread(path)
    orig_img = cv2.cvtColor(orig_img, cv2.COLOR_BGR2RGB)
    cv2.imwrite(temp_filename, orig_img, [cv2.IMWRITE_JPEG_QUALITY, quality])
    compressed_img = cv2.imread(temp_filename)
    diff = SCALE * cv2.absdiff(orig_img, compressed_img)
    return diff
```

Gambar 4.4 Fungsi *compute_ELA_Cv*.

4. Fungsi ELA

Fungsi ini menentukan apakah menggunakan gambar asli atau menjalankan ELA tergantung pada nilai kualitas (*quality*). Jika kualitas adalah 100, fungsi mengembalikan gambar asli. Jika tidak, fungsi memanggil *compute ELA Cv* untuk menghitung ELA dan mengonversi hasilnya kembali menjadi objek gambar menggunakan PIL. Untuk lebih jelas dapat dilihat pada gambar 4.5.

```
def ela(image_path, quality):
    if quality == 100:
        return Image.open(image_path)
    ela_image = compute_ela_cv(image_path, quality)
    ela_image = Image.fromarray(ela_image.astype('uint8'))
    return ela_image
```

Gambar 4.5 Fungsi ELA.

5. Fungsi main

Fungsi ini melakukan hal berikut yaitu Menentukan path gambar yang akan dianalisis, Mendefinisikan berbagai tingkat kualitas JPEG untuk dianalisis, Membuat *subplot* 2x3 untuk menampilkan hasil ELA pada berbagai tingkat kualitas, Untuk setiap tingkat kualitas,

menghitung ELA dan menampilkannya pada *subplot*, Menyesuaikan *layout subplot* agar tampilan rapi dan menampilkan *plot*. Untuk lebih jelas dapat dilihat pada gambar 4.6.

```
def main():
    image_path = "uji/Tp.D.CRN.M.N_nat00083_anil0101_10096.tif" # Ganti dengan path gambar Anda
    quality_levels = [100, 97, 94, 91, 88, 85]
    fig, axs = plt.subplots(2, 3, figsize=(15, 10)) # Membuat subplot 2x3

    for i, quality in enumerate(quality_levels):
        row = i // 3
        col = i % 3
        ela_image = ela(image_path, quality)
        axs[row, col].imshow(ela_image)
        axs[row, col].set_title(f'Quality {quality}')

    plt.tight_layout()
    plt.show()
```

Gambar 4.6 Fungsi Main.

6. Eksekusi Program

Bagian ini memastikan bahwa fungsi main dijalankan saat *script* dijalankan sebagai program utama. Untuk lebih jelas dapat dilihat pada gambar 4.7.

```
if __name__ == "__main__":
    main()
```

Gambar 4.7 Eksekusi Program.

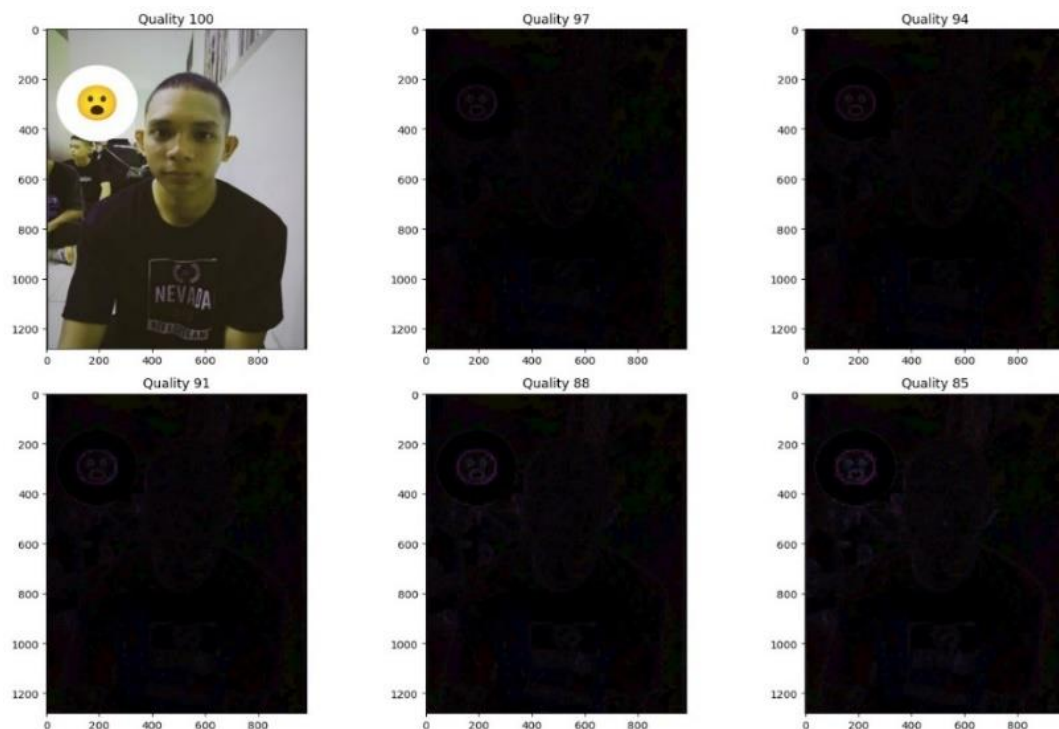
Berikut adalah proses dari implementasi penerapan *Error Level Analysis* (ELA) berdasarkan foto-foto *dataset*.

Tabel 4.1 Implementasi penerapan *Error Level Analysis* (ELA).

No	Penjelasan Proses	Penjelasan
1	Memuat Gambar Asli	Gambar asli dibaca dan dikonversi ke RGB.
2	Menyimpan Ulang dengan Kualitas Berbeda	Gambar asli disimpan ulang dengan tingkat kompresi JPEG yang berbeda (ditentukan oleh nilai kualitas).
3	Menghitung Perbedaan	Perbedaan absolut antara gambar asli dan gambar yang dikompres dihitung dan ditingkatkan skalanya untuk memperjelas perbedaan.
4	Visualisasi Hasil	Hasil ELA untuk berbagai tingkat kompresi ditampilkan dalam <i>subplot</i> .

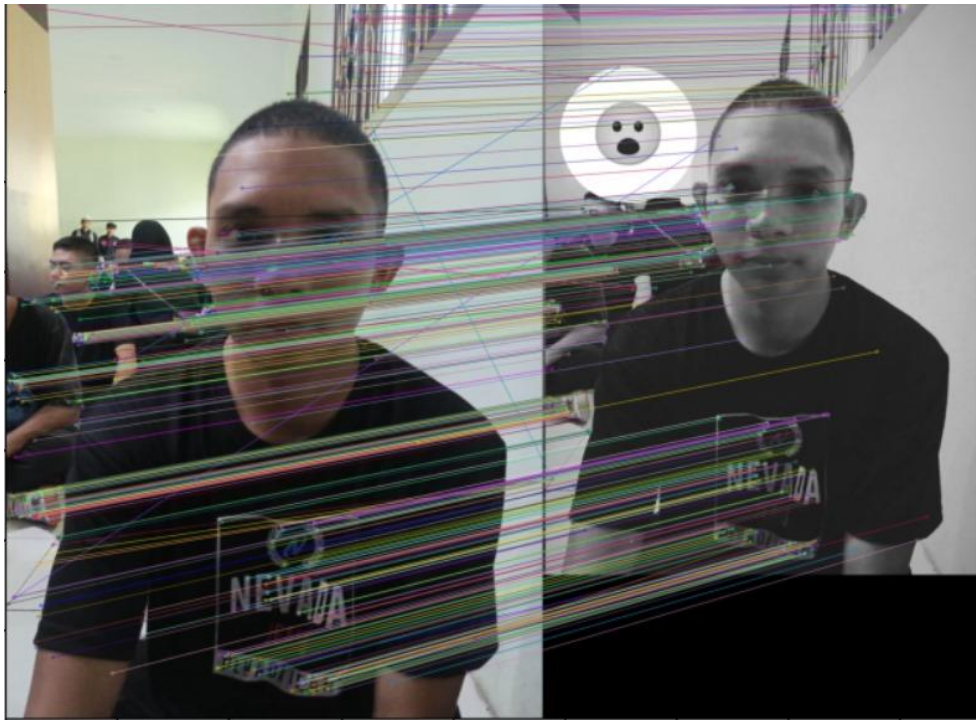
Program tersebut menggunakan dua gambar yang diambil dari sistem file untuk dibandingkan dan dianalisis. Dengan menghitung *Error Level Analysis* (ELA), *Mean Squared Error* (MSE), dan *Peak Signal-to-Noise Ratio* (PSNR), program ini bertujuan untuk mengidentifikasi gambar palsu atau yang telah mengalami manipulasi. Selain itu, program ini menerapkan teknik pengolahan gambar seperti konversi warna, reduksi *noise*, dan normalisasi histogram untuk mempersiapkan gambar sebelum dilakukan perbandingan. Dengan menggunakan gambar-gambar ini sebagai *dataset*, program memberikan hasil analisis yang dapat membantu dalam mendeteksi keaslian atau keabsahan suatu gambar dalam konteks forensik digital.

Berikut merupakan foto yang telah mengalami manipulasi dan kemudian dianalisis menggunakan *Error Level Analysis* (ELA) pada foto tersebut. Untuk lebih jelas dapat dilihat pada gambar 4.8.



Gambar 4.8 Analisi Foto Menggunakan ELA.

Selanjutnya foto akan dianalisis kembali menggunakan *Scale-Invariant Feature Transform* (SIFT) terhadap foto yang terdeteksi adanya manipulasi. Untuk lebih jelas dapat dilihat pada gambar 4.9.



Gambar 4.9 Analisi Foto Menggunakan SIFT.

4.3. Implementasi Sistem

Implementasi sistem merupakan tahap penerapan sistem yang telah didesain atau dirancang, sehingga sistem yang telah dibuat dapat dioperasikan. Sistem ini dibuat dengan menggunakan *android studio* sebagai dasar pembuatan aplikasi *android* menggunakan Bahasa pemrograman *Kotlin*. Berikut ini hasil implementasi perangkat lunak.

a. Tampilan Awal

Tampilan awal adalah halaman yang muncul ketika sebuah aplikasi pertama kali dibuka. Halaman ini akan tampil selama beberapa detik sebelum tampilan utama aplikasi diperlihatkan kepada pengguna. Dapat dilihat pada gambar 4.10.



Deteksi Keaslian Gambar



Gambar 4.10 Tampilan Awal Aplikasi.

b. Tampilan Utama

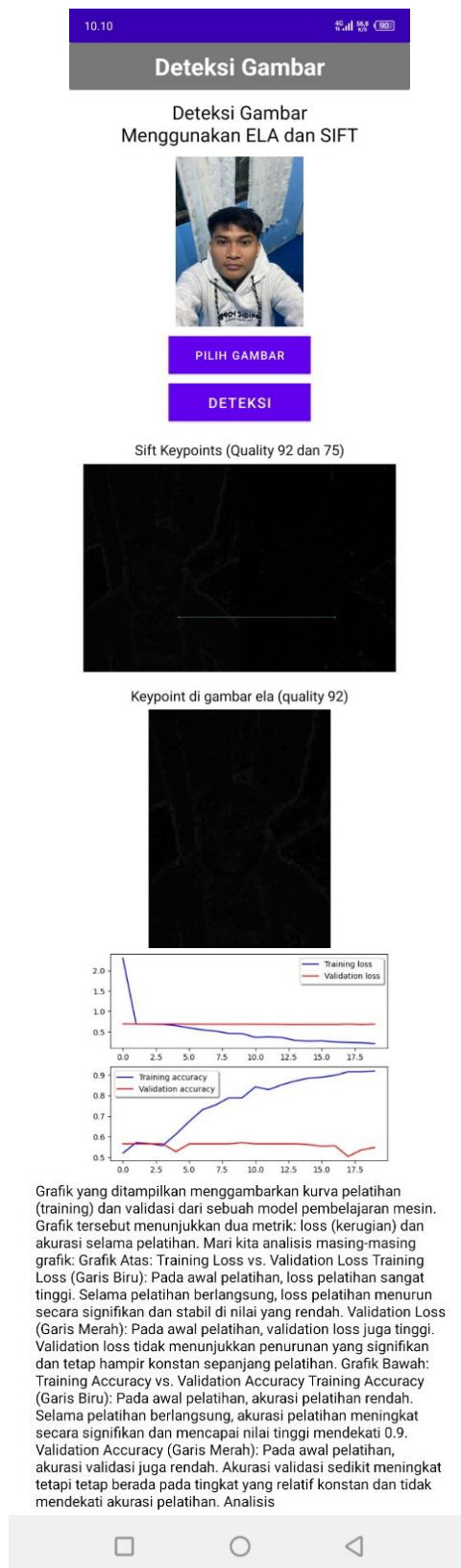
Tampilan Utama merupakan halaman setelah tampilan awal, atau dapat disebut sebagai halaman utama, dimana terdapat app bar yang berisi ikon menu untuk mendeteksi foto. Dapat dilihat pada gambar 4.11.



Gambar 4.11 Tampilan Utama Aplikasi.

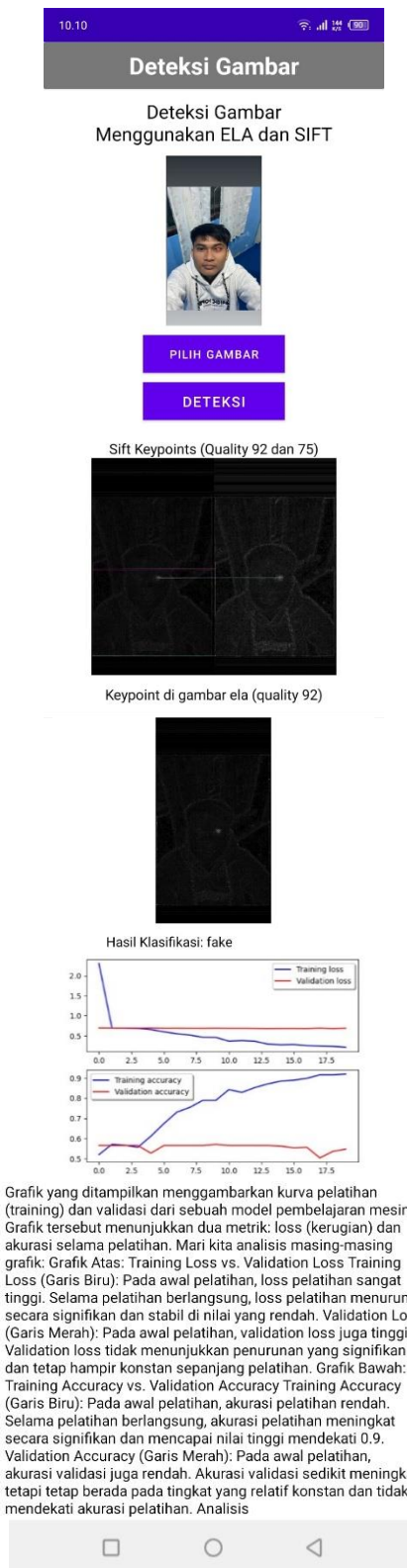
- c. Tampilan hasil foto asli

Tampilan hasil foto asli yang telah di analisis keakurasianya. Dapat dilihat pada gambar 4.12.



Gambar 4.12 Tampilan hasil analisis foto asli.

d. Tampilan hasil foto manipulasi



Gambar 4.13 Tampilan hasil manipulasi.

4.4. Menghubungkan API Dengan Sistem

Pada tahapan ini setelah membuat sistem selanjutnya menghubungkan model yang telah dibuat sebelumnya menggunakan *flask* API ke sistem. Berikut *codingan* dari *flask* API untuk menghubungkan antara sistem dengan model yang telah dibuat. Dapat dilihat pada gambar 4.14.



```

app = Flask(__name__)

# Setup logging
logging.basicConfig(level=logging.DEBUG)

# Muat model yang telah disimpan
model = load_model('model_casia_run1.h5')
sift = cv2.SIFT_create()

def convert_to_ela_image(image, quality):
    temp_file = io.BytesIO()
    ela_file = io.BytesIO()

    image.save(temp_file, 'JPEG', quality=quality)
    temp_file.seek(0)
    temp_image = Image.open(temp_file)

    ela_image = ImageChops.difference(image, temp_image)

    extrema = ela_image.getextrema()
    max_diff = max([ex[1] for ex in extrema])
    if max_diff == 0:
        max_diff = 1
    scale = 255.0 / max_diff

    ela_image = ImageEnhance.Brightness(ela_image).enhance(scale)
    logging.debug(f"ELA image created with quality {quality}")
    return ela_image

```

Gambar 4.14 Menghubungkan API Dengan Sistem.

4.5. Hasil Analisis

Dari hasil analisis akan dibahas mengenai beberapa hal penting dari penelitian yang dilakukan sehingga menjadi acuan penelitian berikut dengan menggunakan metode yang sama.

Metode yang digunakan adalah *Error Level Analysis* (ELA) dan *Scale-Invariant*

Feature Transform (SIFT) memiliki akurasi sebesar 0.9962 atau 99.62% dan untuk pengujian aplikasinya setiap langkah atau halaman dalam aplikasi dari beranda hingga keluar berfungsi sesuai dengan rencana. Dapat dilihat pada tabel 4.2 dan 4.3.

Tabel 4.2 epoch 1/10.

No	Akurasi	Loss	Akurasi Validasi	Loss Validasi
1	0.5212	4.6126	0.5656	0.6926

Tabel 4.3 epoch 10/10.

No	Akurasi	Loss	Akurasi Validasi	Loss Validasi
1	0.7787	0.4773	0.5627	0.6906

Model dimulai dengan akurasi awal yang lebih rendah sebesar 0.5212 dibandingkan dengan proses lainnya. Pada akhir 10 epoch, akurasi meningkat menjadi 0.7787. Akurasi validasi relatif stabil, dimulai dari 0.5656 dan berakhir di 0.5627, menunjukkan proses pelatihan yang lebih stabil dengan lebih sedikit *overfitting* dibandingkan dengan dua proses lainnya untuk lebih jelas dapat dilihat pada gambar 4.14.

```

Epoch 1/10
C:\Users\Rhaayn\AppData\Roaming\Python\Python310\site-packages\keras\src\trainers\data_adapters\py_dataset_adapter.py:122: Use
self.warn_if_super_not_called()
35/35 ————— 34s 784ms/step - accuracy: 0.5212 - loss: 4.6126 - val_accuracy: 0.5656 - val_loss: 0.6926
Epoch 2/10
35/35 ————— 27s 735ms/step - accuracy: 0.5682 - loss: 0.6876 - val_accuracy: 0.5656 - val_loss: 0.6907
Epoch 3/10
35/35 ————— 27s 721ms/step - accuracy: 0.5680 - loss: 0.6792 - val_accuracy: 0.5656 - val_loss: 0.6910
Epoch 4/10
35/35 ————— 27s 731ms/step - accuracy: 0.6081 - loss: 0.6555 - val_accuracy: 0.4723 - val_loss: 0.6928
Epoch 5/10
35/35 ————— 27s 731ms/step - accuracy: 0.6587 - loss: 0.6162 - val_accuracy: 0.4315 - val_loss: 0.6936
Epoch 6/10
35/35 ————— 27s 730ms/step - accuracy: 0.7005 - loss: 0.5777 - val_accuracy: 0.4402 - val_loss: 0.6945
Epoch 7/10
35/35 ————— 27s 743ms/step - accuracy: 0.6848 - loss: 0.5832 - val_accuracy: 0.5685 - val_loss: 0.6924
Epoch 8/10
35/35 ————— 27s 724ms/step - accuracy: 0.7243 - loss: 0.5569 - val_accuracy: 0.5656 - val_loss: 0.6918
Epoch 9/10
35/35 ————— 27s 725ms/step - accuracy: 0.7548 - loss: 0.4871 - val_accuracy: 0.5656 - val_loss: 0.6905
Epoch 10/10
35/35 ————— 27s 745ms/step - accuracy: 0.7787 - loss: 0.4773 - val_accuracy: 0.5627 - val_loss: 0.6906

```

Gambar 4.15 Proses *Epoch* ke-10.

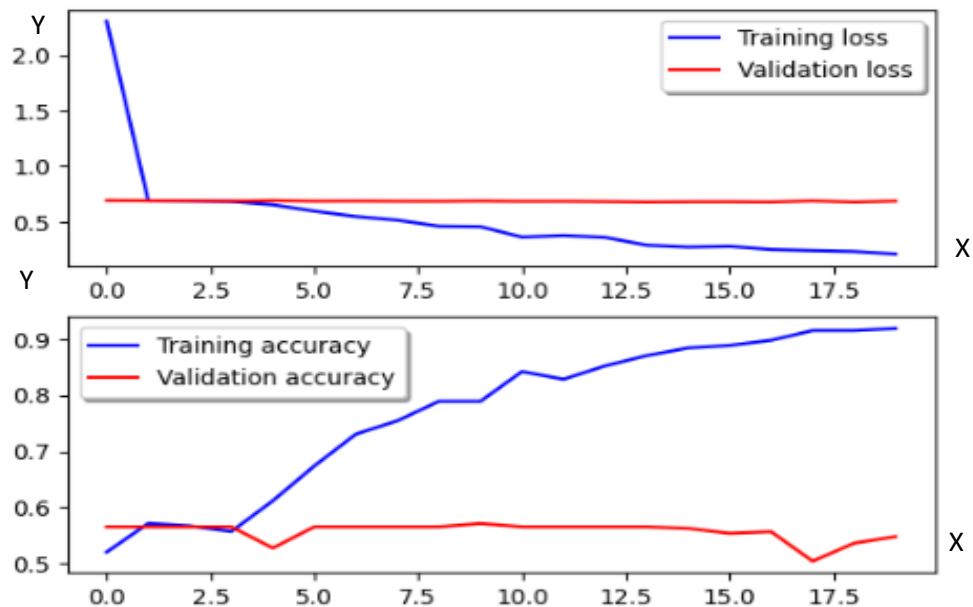
Penjelasan gambar 4.15:

Dalam pengujian model machine learning untuk mendeteksi foto asli dan manipulasi,

biasanya data akan dibagi menjadi dua kelompok utama: data pelatihan dan data validasi.

Proses ini dapat dijelaskan sebagai berikut:

1. Persiapan Data:
 - a. Foto Asli: Kumpulan gambar yang tidak mengalami manipulasi digital.
 - b. Foto Manipulasi: Kumpulan gambar yang telah dimodifikasi menggunakan berbagai teknik manipulasi seperti penambahan objek, penghapusan objek, perubahan warna, dll.
2. Pelatihan Model:
 - a. Model dilatih menggunakan data pelatihan yang terdiri dari foto asli dan manipulasi.
 - b. Selama pelatihan, model belajar untuk membedakan antara fitur-fitur foto asli dan yang telah dimanipulasi.
3. Validasi Model:
 - a. Setelah pelatihan, model diuji menggunakan data validasi yang tidak digunakan selama pelatihan.
 - b. Hasil validasi memberikan gambaran tentang seberapa baik model dapat menggeneralisasi pengetahuannya ke data baru yang belum pernah dilihat sebelumnya.
4. Evaluasi:
 - a. Akurasi dan loss pada data pelatihan dan validasi dibandingkan untuk mengevaluasi performa model.
 - b. Jika akurasi validasi tinggi dan *loss* rendah, model dianggap memiliki performa baik.
 - c. Jika akurasi validasi rendah dan *loss* tinggi, model mungkin *overfitting* atau *underfitting*. Dapat dilihat pada gambar 4.16.



Gambar 4.16 Kurva akurasi dan kurva loss.

Dapat dilihat pada gambar diatas bahwa akurasi terbaik didapatkan pada *epoch* ke-24. Nilai *Validation loss* setelah *epoch* ke-24 mulai datar dan akhirnya meningkat, yang merupakan tanda dari *overfitting*. Metode Identifikasi jumlah *epoch* yang baik untuk digunakan pada saat pelatihan adalah *early stopping*. Dengan metode ini, pelatihan akan dihentikan Ketika nilai akurasi validasi mulai menurun atau nilai *validation loss* mulai meningkat. Dapat dilihat pada tabel 4.4 dan 4.5.

Tabel 4.4 epoch 1/24.

No	Akurasi	Loss	Akurasi Validasi	Loss Validasi
1	0.3480	0.3491	0.5773	0.6929

Tabel 4.5 epoch 24/24.

No	Akurasi	Loss	Akurasi Validasi	Loss Validasi
1	0.9587	0.1427	0.4752	0.7419

```

epochs = 24
batch_size = 32
init_lr = 1e-4
# Define a learning rate schedule function
def lr_schedule(epoch):
    lr = init_lr * np.exp(-epoch / epochs)
    return lr
optimizer = Adam(learning_rate=init_lr)
lr_scheduler = LearningRateScheduler(lr_schedule)
patience = 10
early_stopping = EarlyStopping(monitor='val_accuracy', patience=patience, restore_best_weights=True)
x_train2 = np.array(X_train, copy=True)
y_train2 = np.array(Y_train, copy=True)

datagen = ImageDataGenerator(
    featurewise_center=True,
    featurewise_std_normalization=True,
    rotation_range=10,
    fill_mode='nearest',
    validation_split=0.2
)

datagen.fit(X_train)
print(type(X_train))
validation_generator = datagen.flow(x_train2, y_train2, batch_size=32, subset='validation')
train_generator = datagen.flow(x_train2, y_train2, batch_size=32, subset='training')
history = model.fit(train_generator, epochs=epochs, validation_data=(X_val, Y_val), verbose=1, callbacks=[early_stopping])
✓ 6m 53.2s

```

```

<class 'numpy.ndarray'>
Epoch 1/24
35/35 ————— 31s 852ms/step - accuracy: 0.8480 - loss: 0.3491 - val_accuracy: 0.5773 - val_loss: 0.6929
Epoch 2/24
35/35 ————— 32s 883ms/step - accuracy: 0.8672 - loss: 0.3399 - val_accuracy: 0.5510 - val_loss: 0.6955
Epoch 3/24
35/35 ————— 31s 832ms/step - accuracy: 0.8568 - loss: 0.3377 - val_accuracy: 0.4869 - val_loss: 0.6952
Epoch 4/24
35/35 ————— 30s 834ms/step - accuracy: 0.9022 - loss: 0.2650 - val_accuracy: 0.5190 - val_loss: 0.6962
Epoch 5/24
35/35 ————— 31s 854ms/step - accuracy: 0.9189 - loss: 0.2558 - val_accuracy: 0.4810 - val_loss: 0.7037
Epoch 6/24
35/35 ————— 31s 840ms/step - accuracy: 0.9253 - loss: 0.2099 - val_accuracy: 0.4810 - val_loss: 0.7049
Epoch 7/24
35/35 ————— 32s 872ms/step - accuracy: 0.9196 - loss: 0.2413 - val_accuracy: 0.4840 - val_loss: 0.7116
Epoch 8/24
35/35 ————— 42s 1s/step - accuracy: 0.9054 - loss: 0.1927 - val_accuracy: 0.4927 - val_loss: 0.7102
Epoch 9/24
35/35 ————— 48s 1s/step - accuracy: 0.9188 - loss: 0.1884 - val_accuracy: 0.4956 - val_loss: 0.7103
Epoch 10/24
35/35 ————— 49s 1s/step - accuracy: 0.9388 - loss: 0.1689 - val_accuracy: 0.4840 - val_loss: 0.7281
Epoch 11/24
35/35 ————— 51s 1s/step - accuracy: 0.9587 - loss: 0.1427 - val_accuracy: 0.4752 - val_loss: 0.7419

```

Gambar 4.17 Proses *epoch* ke-24.

Model menunjukkan peningkatan akurasi pelatihan dari 0.8480 menjadi 0.9587. Akurasi validasi berfluktuasi, mencapai puncaknya sekitar 0.5773 tetapi berakhir di 0.4752, yang menunjukkan kemungkinan *overfitting* karena model berkinerja jauh lebih baik pada data pelatihan dibandingkan data validasi.

Dari hasil ini, dapat dilihat bahwa akurasi pelatihan terus meningkat dan *loss* pelatihan terus menurun seiring dengan bertambahnya *epoch*, menunjukkan bahwa model belajar dengan baik. Namun, akurasi validasi menunjukkan fluktuasi dan cenderung menurun, sementara *loss* validasi cenderung naik. Ini mungkin menandakan bahwa model mengalami *overfitting*, di mana model belajar sangat baik pada data pelatihan tetapi tidak dapat

menggeneralisasi dengan baik pada data validasi. Dapat dilihat pada tabel 4.6 dan 4.7.

Tabel 4.6 epoch 1/30.

No	Akurasi	Loss	Akurasi Validasi	Loss Validasi
1	0.8390	0.3832	0.5306	0.6937

Tabel 4.7 epoch 30/30.

No	Akurasi	Loss	Akurasi Validasi	Loss Validasi
1	0.9280	0.2097	0.4985	0.7269

Berikut akurasi yang didapatkan dibawa merupakan hasil proses *epoch* ke-30, dimana *epoch* ke-30 menunjukkan hasil akurasi 0,9280 atau 92,80% untuk lebih jelas dapat dilihat pada gambar 4.17.

Epoch 1/30	31s 786ms/step - accuracy: 0.8390 - loss: 0.3832 - val_accuracy: 0.5306 - val_loss: 0.6937
Epoch 2/30	35/35 27s 740ms/step - accuracy: 0.8702 - loss: 0.3391 - val_accuracy: 0.4810 - val_loss: 0.6942
Epoch 3/30	35/35 27s 739ms/step - accuracy: 0.8764 - loss: 0.3286 - val_accuracy: 0.5306 - val_loss: 0.6932
Epoch 4/30	35/35 27s 732ms/step - accuracy: 0.8756 - loss: 0.2884 - val_accuracy: 0.5248 - val_loss: 0.6904
Epoch 5/30	35/35 26s 719ms/step - accuracy: 0.9079 - loss: 0.2590 - val_accuracy: 0.4636 - val_loss: 0.6950
Epoch 6/30	35/35 26s 718ms/step - accuracy: 0.9161 - loss: 0.2373 - val_accuracy: 0.4810 - val_loss: 0.6977
Epoch 7/30	35/35 27s 739ms/step - accuracy: 0.9177 - loss: 0.2100 - val_accuracy: 0.5015 - val_loss: 0.7026
Epoch 8/30	35/35 26s 722ms/step - accuracy: 0.9263 - loss: 0.2028 - val_accuracy: 0.4956 - val_loss: 0.7048
Epoch 9/30	35/35 26s 718ms/step - accuracy: 0.9355 - loss: 0.1833 - val_accuracy: 0.5160 - val_loss: 0.7141
Epoch 10/30	35/35 27s 729ms/step - accuracy: 0.9390 - loss: 0.1905 - val_accuracy: 0.5160 - val_loss: 0.7334
Epoch 11/30	35/35 27s 731ms/step - accuracy: 0.9280 - loss: 0.2097 - val_accuracy: 0.4985 - val_loss: 0.7269

Gambar 4.18 proses *epoch* ke-30.

Akurasi awal cukup tinggi di 0.8390, dan akurasi pelatihan meningkat menjadi 0.9280. Akurasi validasi dimulai dari 0.5306 dan berakhir di 0.4985. Ada perbedaan mencolok antara kinerja pelatihan dan validasi, yang menunjukkan kemungkinan *overfitting*.

Hasil ini menunjukkan bahwa model mungkin mengalami beberapa tingkat *overfitting*, di mana model belajar sangat baik pada data pelatihan tetapi tidak sepenuhnya mampu menggeneralisasi pengetahuannya pada data validasi. Ini ditandai dengan peningkatan

akurasi pelatihan yang konsisten dan penurunan *loss* pelatihan, tetapi dengan fluktuasi dan kurangnya tren penurunan pada *loss* validasi.

Kesimpulan dari proses ke 3 *epoch* di atas adalah sebagai berikut:

1. Proses Pertama:

Peningkatan signifikan dalam akurasi pelatihan dengan fluktuasi tinggi dalam akurasi validasi, menunjukkan *overfitting*.

2. Proses Kedua:

Akurasi awal tinggi, tetapi akurasi validasi tidak meningkat secara signifikan, menunjukkan *overfitting*.

3. Proses Ketiga:

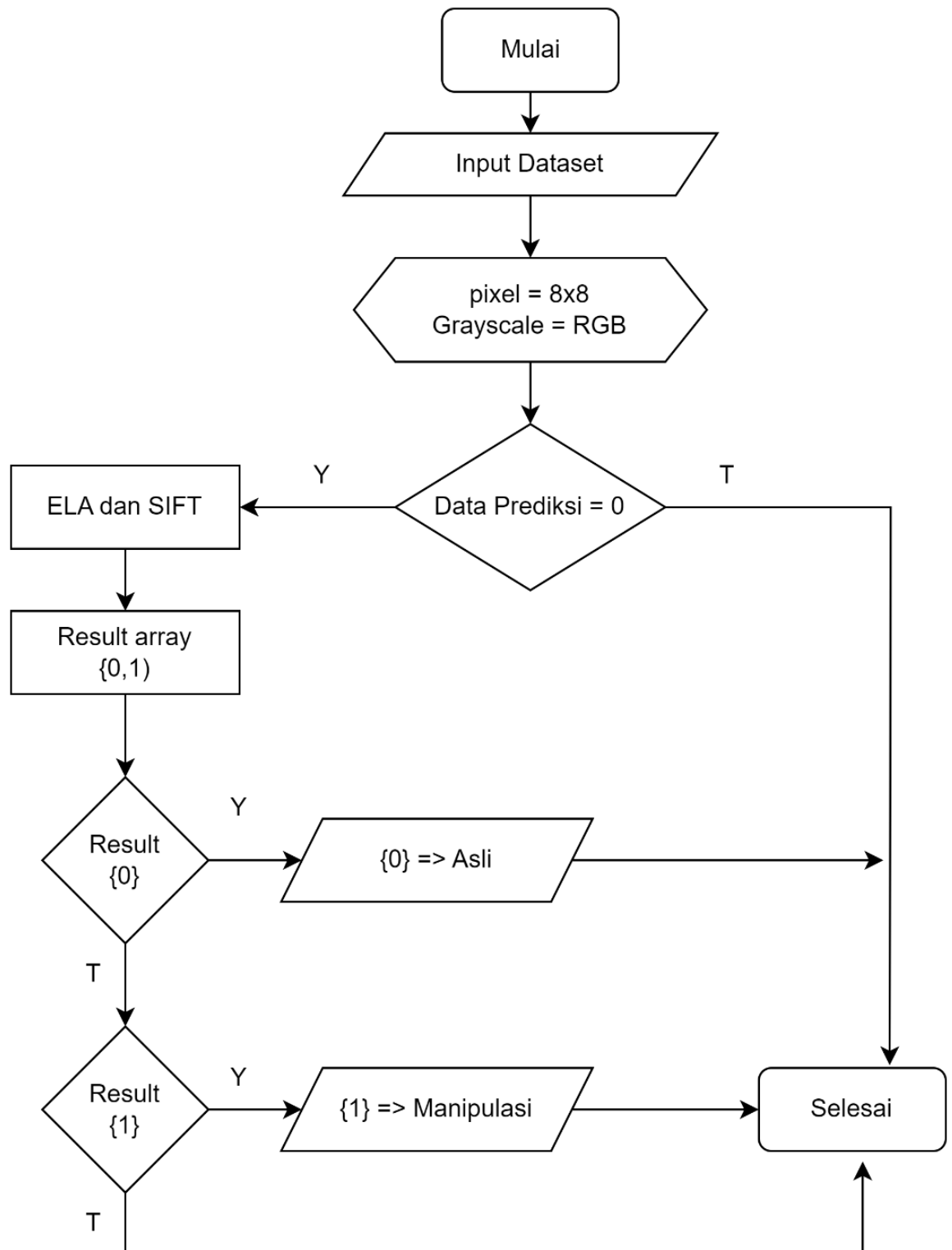
Akurasi validasi stabil dengan peningkatan moderat dalam akurasi pelatihan, menunjukkan proses pelatihan yang lebih seimbang.

Perbedaan antara kinerja pelatihan dan validasi pada setiap *epoch* menunjukkan pentingnya memonitor dan mengatasi *overfitting* dalam model pembelajaran mendalam.

4.6. White Box

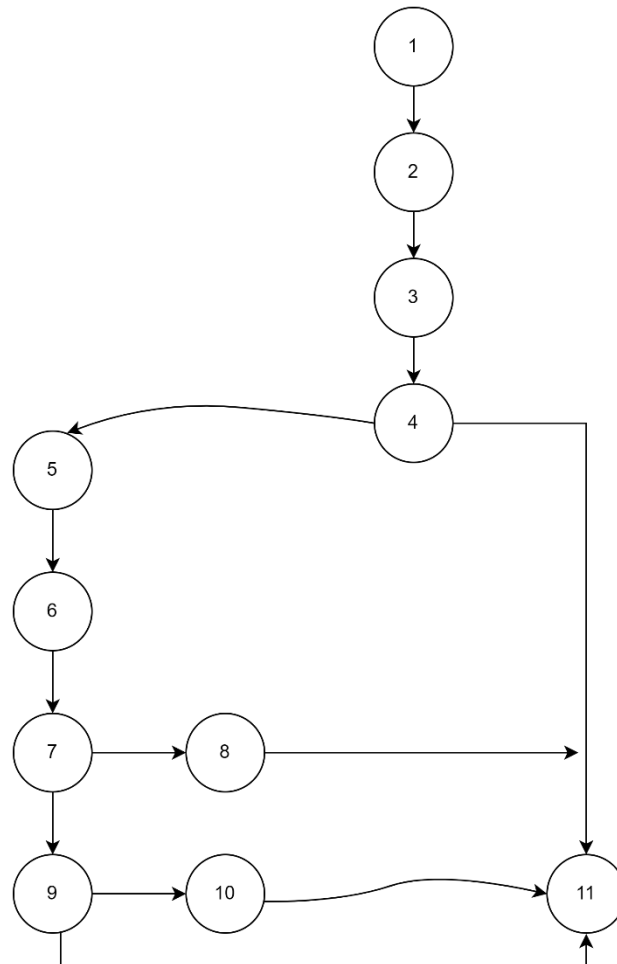
Pada bagian ini dilakukan pengujian menggunakan metode *white box*. Bagian yang akan diuji adalah prediksi data *testing*. Berikut merupakan Langkah-langkah pengujian prediksi data *testing*.

1. Menentukan *flowchart* pengujian, untuk lebih jelasnya dapat dilihat pada gambar 4.19.



Gambar 4.19 Flowchart Data Prediksi.

2. Menbuat *flowgrap* pengujian, untuk lebih jelasnya dapat dilihat pada gambar 4.20.



Gambar 4.120 *Flowgrap* Pengujian Data Prediksi.

3. Setelah menentukan *flowgraph* maka selanjutnya mencari *Cyclomatic Complexity* *grap* alur G.

$$\begin{aligned}
 V(G) &= (E - N) + 2 \\
 &= (13 - 11) + 2 \\
 &= 2 + 2 = 4
 \end{aligned}$$

4. Menentukan *Independent part*.

Part 1 = 1-2-3-4-5-6-7-8-11

Part 2 = 1-2-3-4-5-6-7-9-10-11

Part 3 = 1-2-3-4-5-6-7-9-11

Part 4 = 1-2-3-4-11

Berikut penjelasan dari pengujian *white box* diatas dengan Proses pengujian *white box* dilakukan dengan langkah-langkah sebagai berikut:

1. Analisis Struktur Kode:
 - a. Memahami setiap *node* dalam diagram yang mewakili pernyataan atau blok kode spesifik dalam program.
 - b. Identifikasi titik awal (*node* 1) dan titik akhir (*node* 11).
2. Penentuan Jalur Pengujian:
 - a. Mengidentifikasi semua jalur yang mungkin dilalui dalam diagram alir.
 - b. Jalur utama yang mungkin antara lain:
 - c. $1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 6 \rightarrow 7 \rightarrow 8 \rightarrow 11$
 - d. $1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 6 \rightarrow 7 \rightarrow 9 \rightarrow 10 \rightarrow 11$
 - e. $1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 11$ (menggunakan *loop* kembali dari 11 ke 4).
3. Definisi Kasus Uji:
 - a. Membuat kasus uji untuk memastikan semua jalur di diagram tercakup. Ini termasuk pengujian kondisi di *node* 5 dan *node* 10 yang memiliki jalur bercabang.
 - b. Contoh kasus uji mungkin meliputi skenario di mana kondisi pada *node* 5 membawa aliran ke *node* 6, dan kondisi pada *node* 10 membawa aliran kembali ke *node* 4.
4. Implementasi Kasus Uji:
 - c. Mengimplementasikan tes berdasarkan jalur yang telah diidentifikasi.
 - d. Setiap kasus uji harus memastikan bahwa alur kontrol mengikuti jalur yang ditentukan

ditentukan, memverifikasi bahwa setiap pernyataan dan kondisi dieksekusi. Eksekusi Tes: Menjalankan setiap kasus uji dan memverifikasi bahwa hasilnya sesuai dengan yang diharapkan.

- e. Menangkap hasil eksekusi untuk memastikan bahwa semua jalur dan cabang dalam diagram telah diuji.

5. Analisis Hasil Tes:

- a. Menilai hasil eksekusi untuk mengidentifikasi potensi kesalahan atau cacat dalam logika program.
- b. Jika ditemukan kesalahan, analisis untuk memahami sumber kesalahan dan memperbaikinya.

6. *Regression Testing*:

Setelah perbaikan dilakukan, menjalankan kembali tes untuk memastikan bahwa perubahan tidak memperkenalkan kesalahan baru dan jalur yang telah diuji sebelumnya masih valid.

BAB V

PENUTUP

5.1. Kesimpulan

Penelitian ini bertujuan untuk mengatasi masalah manipulasi foto digital yang semakin marak terjadi, terutama yang menggunakan teknik *copy-move* untuk membuat foto palsu atau melakukan penipuan. Masalah utama yang dihadapi adalah bagaimana mengidentifikasi dan mendeteksi keaslian foto digital yang telah dimanipulasi.

Untuk menyelesaikan masalah ini, penelitian ini menggunakan dua metode utama:

1. *Error Level Analysis* (ELA): Metode ini digunakan untuk mendeteksi ketidakkonsistenan pada tingkat kompresi gambar yang dapat mengindikasikan adanya perubahan atau manipulasi. ELA bekerja dengan mengidentifikasi area gambar yang memiliki tingkat kesalahan berbeda, yang biasanya terjadi akibat penyuntingan.
2. *Scale-Invariant Feature Transform* (SIFT): Metode ini digunakan untuk mengidentifikasi dan mencocokkan titik-titik kunci pada gambar. SIFT mendeteksi fitur-fitur khusus dalam citra yang dapat digunakan untuk mendeteksi area yang telah diduplikasi dan dipindahkan.

Hasil dari penelitian ini menunjukkan bahwa kombinasi metode ELA dan SIFT efektif dalam mendeteksi manipulasi foto digital. Implementasi ELA mampu mengidentifikasi perubahan tingkat kesalahan dalam gambar dengan baik, sementara SIFT dapat mendeteksi fitur-fitur kunci yang telah dimanipulasi. Tingkat akurasi dari kombinasi kedua metode ini cukup tinggi, menjadikannya alat yang berguna dalam forensik digital untuk mendeteksi manipulasi foto. Dengan demikian, penelitian ini memberikan kontribusi yang

signifikan dalam meningkatkan pemahaman dan kemampuan untuk mendeteksi keaslian foto digital, sehingga dapat membantu mencegah penyebaran informasi palsu dan penipuan melalui gambar digital.

5.2. Saran

Adapun saran yang dapat diberikan untuk mengembangkan penelitian ini adalah sebagai berikut:

1. Penelitian selanjutnya dapat menggunakan lebih banyak jenis foto asli dan manipulasi, untuk lebih maksimal dalam menganalisis foto asli dan manipulasi agar mendapatkan akurasi yang maksimal.
2. Untuk penelitian selanjutnya disarankan untuk menggunakan *dataset* yang lebih banyak lagi, agar dapat meningkatkan akurasi yang lebih baik lagi nantinya.
3. Dan menggunakan metode yang berbeda dengan penelitian ini agar dapat mengetahui hasil dan kualitas dari masing-masing metode.

DAFTAR PUSTAKA

- Ady Saputro, I. 2022. Forensika Citra Digital Untuk Menganalisis Kecocokan Objek Menggunakan Metode SIFT. *Jurnal Teknik Informatika Dan Sistem Informasi (JATISI)*, ISSN:2407-4322, Vol. 9 Issue. 4 Desember, 2022
- Bisri, H., & Marzuki, M. I. 2023. Forensik Citra Digital Menggunakan Metode *Error Level Analysis*, *Clone Detection* dan *Exif* Untuk Deteksi Keaslian Gambar. *G-Tech: Jurnal Teknologi Terapan*, E-ISSN:2623-064X, Vol. 7 Issue. 2 April, 2023
- Candra, P. N., & Prapanca, A. 2020. Klasifikasi Gambar Asli dan Manipulasi Menggunakan *Error Level Analysis* (ELA) Sebagai Proses Komputasi Metode *Convolutional Neural Network* (CNN). *Journal of Informatics and Computer Science (JINACS)*, ISSN:2686-2220, Vol. 2 Issue. 1 Oktober, 2020
- Charis Noiija, H., Wemaf, P. A., Nurdianty, O. A., Sohilait, W., Haumahu, S., Yusuf, H., Tomagola, K., Rindi, W., Ana, L., Salaiswa, T. S., Akuntansi, J., Ekonomi, F., & Bisnis, D. 2023. Perancangan Sistem Informasi Akuntansi Untuk Analisis Siklus Pendapatan Pada Orantata Celullar Menggunakan DFD Dan Flowchart. E-ISSN:2986-6766, Vol. 1 Issue. 2 Juni, 2023
- Hari, M., Arta, K., Kusuma, I. G. B., Rial, M., & Askar, R. 2020. Analisis Citra Digital Dengan *Tools Image Forensic* Berupa *Fotoforensic*, *Jpegsnoop*, *Ghiro*, Dan *Forensically*. 16. Vol. 2 Issue. 9 Juli, 2020
- Khairan, A., Muin, Y., & Mamonto, A. M. 2023. *Improved Accuracy For Forensic Investigation Using SIFT-Based Copy-Move Forgery Detection*. ISSN: 2407-4322, Vol. 10 Issue. 4 Desember, 2023
- Nova, S. H., Widodo, A. P., & Warsito, B. 2022. Analisis Metode *Agile* pada Pengembangan Sistem Informasi Berbasis *Website: Systematic Literature Review*. Vol. 21 Issue. 1 Januari, 2021
- Novianti, E., Amanda Tahalea, S., & Adhi Purnama, P. 2023. Pelatihan Penulisan Deskripsi Karya Foto Yang Diaplikasikan di Media Sosial Instagram. *Jurnal Ilmu Pendidikan Non Formal*, E-ISSN:2721-7310, Vol. 09 Issue. 2 Mei, 2023
- Prabowo, D. D. A., & Wardoyo, D. T. W. 2022. Evaluasi *Library Automation System Professional* (LIANSPRO) di Perpustakaan. *Journal of Education and Instruction*

(JOEAI), E-ISSN:2614-8617, Vol. 5 Issue. 1 Juni, 2022

- Praniffa, A. C., Syahri, A., Sandes, F., Fariha, U., Giansyah, Q. A., & Hamzah, M. L. 2023. Pengujian *Black Box* Dan *White Box* Sistem Informasi Parkir Berbasis Web *Black Box and White Box Testing of Web-Based Parking Information System*. Jurnal Testing Dan Implementasi Sistem Informasi, Vol. 1 Issue. 1 Febuari, 2023
- Pratasik, S., & Rianto, I. 2020. Pengembangan Aplikasi E-DUK Dalam Pengelolaan SDM Menggunakan Metode *Agile Development*. *CogITo Smart Journal*, Vol. 6 Issue. 2 Desember, 2020
- Purnama, K. E., Rozikin, C., & Ridha, A. A. 2023. Analisis Forensic Citra Digital Menggunakan Teknik *Error Level Analysis* Dan Metadata Berdasarkan Metode Nist. Jurnal Mahasiswa Teknik Informatika (JATI), Vol. 7 Issue. 2 April, 2023
- Rejeki, M. S., & Tarmuji, A. 2013. Membangun aplikasi *autogenerate script* ke *Flowchart* untuk mendukung *business process Reengineering*. Jurnal Sarjana Teknik Informatika, E-ISSN:2338-5197, Vol. 1 Issue. 2 Oktober, 2013
- Salin-pindahkan, D. P., Muin, Y., & Mamonto, A. M. 2023. Peningkatan Akurasi Investigasi Forensik Menggunakan Berbasis *SIFT*. 7, 1–7. ISSN:2614-5278, Vol.7 Issue. 10 Oktober, 2023
- Singh, R., Verma, S., Yadav, S. A., & Vikram Singh, S. 2022. *Copy-move Forgery Detection using SIFT and DWT detection Techniques*. *Proceedings of 3rd International Conference on Intelligent Engineering and Management, ICIEM 2022, January 2023*, Vol. 3 Issue. 4 April, 2022
- Suhailie. 2021. Mendeteksi Gambar yang Dimanipulasi dengan Metode *Image Processing*. Vol. 2 Issue. 10 Juli, 2021
- Waruwu, M. 2023. Pendekatan Penelitian Pendidikan: Metode Penelitian Kualitatif, Metode Penelitian Kuantitatif dan Metode Penelitian Kombinasi (*Mixed Method*). Jurnal Pendidikan Tambusai, ISSN:2614-6754, Vol. 7 Issue. 1 Januari, 2023
- Wicaksono, A., Mardiyantoro, N., & Sibyan, H. 2022. Penerapan Metode *Error Level Analysis* Untuk Mendeteksi Modifikasi Citra Digital. BINER Jurnal Ilmiah Informatika Dan Komputer, E-ISSN:2828-0229, Vol.1 Issue. 1 Januari, 2022