

**DECISION SUPPORT SYSTEM UNTUK MENENTUKAN TINGKAT
KERUSAKAN BANGUNAN MENGGUNAKAN
METODE PCA - CLUSTERING**

SKRIPSI

Oleh :
AULIA ILMI MAGHFIRA RIDWAN
NIM. 210605110158



**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2024**

***DECISION SUPPORT SYSTEM UNTUK MENENTUKAN TINGKAT
KERUSAKAN BANGUNAN MENGGUNAKAN
METODE PCA - CLUSTERING***

SKRIPSI

Diajukan kepada:
Universitas Islam Negeri Maulana Malik Ibrahim Malang
Untuk memenuhi Salah Satu Persyaratan dalam
Memperoleh Gelar Sarjana Komputer (S.Kom)

Oleh :
AULIA ILMI MAGHFIRA RIDWAN
NIM. 210605110158

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2024**

***DECISION SUPPORT SYSTEM UNTUK MENENTUKAN TINGKAT
KERUSAKAN BANGUNAN MENGGUNAKAN
METODE PCA - CLUSTERING***

SKRIPSI

Diajukan kepada:
Universitas Islam Negeri Maulana Malik Ibrahim Malang
Untuk memenuhi Salah Satu Persyaratan dalam
Memperoleh Gelar Sarjana Komputer (S.Kom)

Oleh :
AULIA ILMI MAGHFIRA RIDWAN
NIM. 210605110158

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2024**

HALAMAN PERSETUJUAN

DECISION SUPPORT SYSTEM UNTUK MENENTUKAN TINGKAT KERUSAKAN BANGUNAN MENGGUNAKAN METODE PCA - CLUSTERING

SKRIPSI

Oleh :

AULIA ILMI MAGHFIRA RIDWAN
NIM. 210605110158

Telah Diperiksa dan Disetujui untuk Diuji:
Tanggal: 12 Desember 2024

Pembimbing I,



Dr. Agung Teguh Wibowo Almais, M.T
NIP. 19860301 202321 1 016

Pembimbing II,



Ahmad Fahmi Karami, M.Kom
NIP. 19870909 202012 1 001

Mengetahui,

Ketua Program Studi Teknik Informatika
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang



Dr. H. Fachrul Kurniawan, M.MT., IPU
NIP. 19771020 200912 1 001

HALAMAN PENGESAHAN

DECISION SUPPORT SYSTEM UNTUK MENENTUKAN TINGKAT KERUSAKAN BANGUNAN MENGGUNAKAN METODE PCA - CLUSTERING

SKRIPSI

Oleh :
AULIA ILMI MAGHFIRA RIDWAN
NIM. 210605110158

Telah Dipertahankan di Depan Dewan Penguji Skripsi
dan Dinyatakan Diterima Sebagai Salah Satu Persyaratan
Untuk Memperoleh Gelar Sarjana Komputer (S.Kom)
Tanggal: 12 Desember 2024

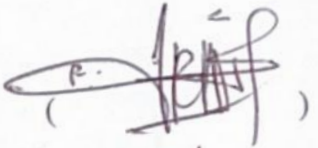
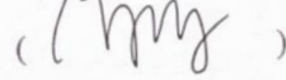
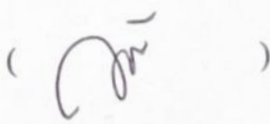
Susunan Dewan Penguji

Ketua Penguji : Dr. Ririen Kusumawati, M.Kom
NIP. 19720309 200501 2 002

Anggota Penguji I : Fajar Rohman Hariri, M. Kom
NIP. 19890515 201801 1 001

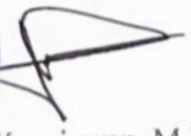
Anggota Penguji II : Dr. Agung Teguh Wibowo Almais, M.T
NIP. 19860301 202321 1 016

Anggota Penguji III : Ahmad Fahmi Karami, M.Kom
NIP. 19870909 202012 1 001

()
()
()
()

Mengetahui dan Mengesahkan,
Ketua Program Studi Teknik Informatika
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang




Dr. Ir. Fachrul Kurniawan, M.MT., IPU
NIP. 19771020 200912 1 001

PERNYATAAN KEASLIAN TULISAN

Saya yang bertanda tangan di bawah ini:

Nama : Aulia Ilmi Maghfira Ridwan
NIM : 210605110158
Fakultas / Program Studi : Sains dan Teknologi / Teknik Informatika
Judul Skripsi : *Decision Support System* Untuk Menentukan
Tingkat Kerusakan Bangunan Menggunakan Metode
PCA – *Clustering*

Menyatakan dengan sebenarnya bahwa Skripsi yang saya tulis ini benar-benar merupakan hasil karya saya sendiri, bukan merupakan pengambil alihan data, tulisan, atau pikiran orang lain yang saya akui sebagai hasil tulisan atau pikiran saya sendiri, kecuali dengan mencantumkan sumber cuplikan pada daftar pustaka.

Apabila dikemudian hari terbukti atau dapat dibuktikan skripsi ini merupakan hasil jiplakan, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Malang, 19 Desember 2024

Yang membuat pernyataan,



Aulia Ilmi Maghfira Ridwan

NIM. 210605110158

MOTTO

“Whatever you are, be a good one”

HALAMAN PERSEMBAHAN

Dengan penuh rasa syukur, karya ini penulis persembahkan kepada:

Ayahanda dan Ibunda tercinta dan adik tersayang

Yang senantiasa memberikan doa, semangat, dan kasih sayang tanpa batas dan selalu menjadi sumber motivasi dan penyemangat.

Seluruh keluarga besar

Atas dukungan dan doa yang mengiringi langkah penulis.

Dosen Pembimbing dan Seluruh Pengajar

Yang telah membimbing serta memberikan ilmu yang berharga.

Teman-teman Aster Teknik Informatika 2021

Yang selalu menemani perjalanan ini dengan kebersamaan dan semangat.

Dan terakhir, untuk diri sendiri

Yang telah berusaha dan bertahan hingga tahap ini.

KATA PENGANTAR

Assalamu'alaikum Warahmatullahi Wabarakatuh

Alhamdulillahirabbil'alamin, segala puji syukur penulis panjatkan ke hadirat Allah SWT. atas limpahan rahmat, nikmat, dan hidayah-Nya, sehingga penulis dapat menyelesaikan tugas skripsi yang berjudul “*Decision Support System Untuk Menentukan Tingkat Kerusakan Bangunan Menggunakan Metode PCA - Clustering*” dengan baik dan lancar. Shalawat serta salam tak lupa penulis sampaikan kepada Nabi Muhammad SAW., sosok teladan mulia yang telah membawa kita dari zaman kegelapan menuju zaman kebenaran, Islam, dan ilmu pengetahuan yang kita rasakan manfaatnya hingga hari ini.

Tugas skripsi ini disusun dalam rangka memenuhi salah satu syarat kelulusan untuk memperoleh gelar Sarjana Komputer pada Program Studi Teknik Informatika, Fakultas Sains dan Teknologi, UIN Maulana Malik Ibrahim Malang. Dalam proses penyusunan tugas ini, penulis mendapatkan bantuan, dukungan, serta doa dari banyak pihak, baik secara langsung maupun tidak langsung. Untuk itu, penulis ingin menyampaikan rasa terima kasih yang sebesar-besarnya kepada:

1. Prof. Dr. M. Zainuddin, M.A., selaku Rektor Universitas Islam Negeri (UIN) Maulana Malik Ibrahim Malang.
2. Prof. Dr. Sri Harini, M.Si., selaku Dekan Fakultas Sains dan Teknologi UIN Maulana Malik Ibrahim Malang.
3. Dr.Ir.Fachrul Kurniawan, M.MT., IPU., selaku Ketua Program Studi Teknik Informatika UIN Maulana Malik Ibrahim Malang.

4. Dr. Agung Teguh Wibowo Almais, M.T., selaku Dosen Pembimbing I, yang dengan kesabaran dan dedikasinya membimbing penulis selama penyusunan skripsi.
5. Ahmad Fahmi Karami, M.Kom., selaku Dosen Pembimbing II, yang senantiasa memberikan arahan dan masukan yang sangat berharga dalam pengerjaan skripsi ini.
6. Dr. Ririen Kusumawati, M.Kom., dan Fajar Rohman Hariri, M.Kom., selaku Dosen Penguji yang telah memberikan kritik, saran, dan bimbingan untuk menyempurnakan skripsi ini.
7. Nia Faricha, S.Si., admin Program Studi Teknik Informatika, yang membantu dalam urusan administrasi dan selalu mengingatkan tentang kelengkapan berkas.
8. Bapak Ridwan dan Ibu Ruliyanti, orang tua tercinta, serta Roby Arinal Haq, adik penulis, yang selalu menjadi pilar kekuatan dengan doa, cinta, dan dukungan tanpa henti.
9. Bintang Miftaqul Huda, sosok yang istimewa bagi penulis yang senantiasa memberikan semangat, motivasi, dan dukungan, baik secara langsung maupun tidak langsung dan menjadi sumber inspirasi di saat penulis membutuhkan.
10. Rekan seperjuangan di Lab Database, yaitu Putri, Ima, Najah, Intan, Roihan, Charles, dan Gigih, yang selalu berbagi pengalaman, tawa, dan semangat selama penelitian berlangsung.

11. Sahabat-sahabat terbaik, Aurell, Jayyid, Galang, Fahri, Hera, Manda, Hanna, Imelda, dan Anindya, yang telah menjadi penyemangat sejak awal kuliah hingga tugas akhir ini.
12. Putri Purnamasari, sahabat terdekat yang tak tergantikan, selalu ada dalam suka dan duka layaknya seorang saudara. Terima kasih telah menjadi pendengar setia dan pemberi semangat sepanjang perjalanan ini.
13. Rekan-rekan Ma'had ABA kamar 02 tahun 2021 dan teman-teman angkatan Aster 2021, yang telah memberikan kebersamaan dan pengalaman berharga selama masa studi.
14. Teman-teman di HIMATIF Encoder, khususnya Divisi Public Relation 2022 dan 2023, yang telah menjadi penyemangat di tengah tantangan akademik.
15. Teman-teman KKM Desa Gondowangi dan rekan-rekan PKL, yang memberikan pengalaman nyata dalam pengabdian kepada masyarakat dan dunia kerja.
16. Seluruh keluarga, teman, sahabat, dan kerabat penulis yang tidak dapat penulis sebutkan satu per satu, yang turut memberikan bantuan, semangat, dukungan, dan doa untuk penulis.
17. Dan terakhir tak lupa terimakasih kepada diri sendiri, yang dengan tekad kuat mampu melewati setiap tantangan, rintangan, dan kesulitan selama perjalanan ini, membuktikan bahwa kerja keras dan usaha tidak pernah sia-sia.

Penulis menyadari bahwa penelitian dalam tugas skripsi ini masih jauh dari kata sempurna dan memiliki berbagai keterbatasan. Oleh karena itu, dengan penuh kerendahan hati, penulis membuka diri untuk menerima kritik dan saran yang

membangun dari para pembaca guna menjadi bahan evaluasi dan pengembangan di masa mendatang. Penelitian ini juga memiliki potensi untuk dilanjutkan dan dikembangkan lebih jauh dalam penelitian berikutnya, sehingga dapat melengkapi kekurangan yang ada. Penulis berharap, karya ini tidak hanya memberikan manfaat bagi pembaca, tetapi juga dapat berkontribusi positif bagi masyarakat secara luas.

Wassalamu'alaikum Warahmatullahi Wabarakatuh.

Malang, 18 November 2024

Penulis

DAFTAR ISI

HALAMAN PENGAJUAN	ii
HALAMAN PERSETUJUAN	iii
HALAMAN PENGESAHAN	iv
PERNYATAAN KEASLIAN TULISAN	v
MOTTO	vi
HALAMAN PERSEMBAHAN	vii
KATA PENGANTAR.....	viii
DAFTAR ISI.....	xii
DAFTAR GAMBAR.....	xiv
DAFTAR TABEL	xv
ABSTRAK	xvi
ABSTRACT	xvii
الملخص	xviii
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	5
1.3 Batasan Masalah	5
1.4 Tujuan Penelitian	5
1.5 Manfaat Penelitian	6
BAB II STUDI PUSTAKA	7
2.1 Penelitian Tentang LBP-PCA	7
2.2 Bencana Alam	9
2.3 <i>Decision Support System (DSS)</i>	10
2.4 <i>Local Binary Pattern (LBP)</i>	12
2.5 <i>Principal Component Analysis (PCA)</i>	14
BAB III METODE PENELITIAN	18
3.1 Desain Penelitian	18
3.2 Pengumpulan Data	19
3.3 Desain Sistem.....	21
BAB IV HASIL DAN PEMBAHASAN	38
4.1 Uji Coba Sistem	38
4.2 Hasil Uji Coba.....	39
4.2.1 Proses LBP.....	39
4.2.2 Proses PCA	54
4.2.3 Augmentasi Data.....	67
4.3 Pembahasan.....	78
4.4 Integrasi Islam.....	86
4.4.1 <i>Muamalah Ma'a Allah</i>	86
4.4.2 <i>Muamalah Ma'a An-Nas</i>	87
4.4.3 <i>Muamalah Ma'a Al-Alam</i>	88
BAB V KESIMPULAN DAN SARAN	89
5.1 Kesimpulan	89
5.2 Saran	89

DAFTAR PUSTAKA

DAFTAR GAMBAR

Gambar 2. 1 Pola Ketetangaan LBP	12
Gambar 3. 1 Desain Penelitian.....	18
Gambar 3. 2 Desain Sistem.....	21
Gambar 3. 3 Tahapan <i>Local Binary Pattern</i>	22
Gambar 3. 4 Proses <i>Grayscale</i>	23
Gambar 3. 5 Histogram LBP.....	31
Gambar 4. 1 Gambar RGB.....	39
Gambar 4. 2 Gambar <i>Grayscale</i>	40
Gambar 4. 3 Gambar <i>resize</i>	40
Gambar 4. 4 Pola Radius.....	42
Gambar 4. 5 Grid 5x5.....	48
Gambar 4. 6 Data Normalisasi	54
Gambar 4. 7 Visualisasi Hasil Kategori.....	66
Gambar 4. 8 Proses Augmentasi Reflection	68
Gambar 4. 9 Hasil Visualisasi Reflection	70
Gambar 4. 10 Rotated	71
Gambar 4. 11 Hasil Visualisasi Rotated	73
Gambar 4. 12 Grayscale Scalling.....	74
Gambar 4. 13 Hasil Zoom.....	75
Gambar 4. 14 Visualisasi Scalling	77

DAFTAR TABEL

Tabel 2. 1 Penelitian Tentang LBP-PCA	8
Tabel 3. 1 Label Alternatif	20
Tabel 3. 2 Perbandingan Nilai Biner.....	26
Tabel 3. 3 Perhitungan Nilai Biner	30
Tabel 4. 1 Hasil Ekstraksi	49
Tabel 4. 2 PC 1.....	58
Tabel 4. 3 Explained Variance Ration	60
Tabel 4. 4 Eigen Value.....	60
Tabel 4. 5 Kategori Kerusakan Bangunan	62
Tabel 4. 6 Hasil PC1 yang sudah dinormalisasi.....	63
Tabel 4. 7 Titik Hasil Rusak	64
Tabel 4. 8 Hasil Ekstraksi Reflection.....	68
Tabel 4. 9 Hasil Titik Reflection.....	69
Tabel 4. 10 Hasil Ekstraksi Rotated.....	71
Tabel 4. 11 Hasil Titik Rotated	73
Tabel 4. 12 Ekstraksi Fitu Scalled.....	75
Tabel 4. 13 Jumlah Titik Scalled.....	77
Tabel 4. 14 Hasil Pengujian	80
Tabel 4. 15 Skor Validasi.....	85

ABSTRAK

Maghfira, Aulia Ilmi. 2024. ***Decision Support System Untuk Menentukan Tingkat Kerusakan Bangunan Menggunakan Metode PCA - Clustering***. Skripsi. Program Studi Teknik Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang. Pembimbing: (I) Dr. Agung Teguh Wibowo Almais, M.T (II) Ahmad Fahmi Karami, M.Kom

Kata Kunci: Kerusakan Bangunan, Bencana Alam, Local Binary Pattern (LBP), Principal Component Analysis (PCA), Image Analysis

Kerusakan bangunan akibat bencana alam seringkali memerlukan penilaian yang cepat dan akurat untuk mendukung langkah-langkah mitigasi. Penelitian ini bertujuan menentukan tingkat kerusakan bangunan pasca bencana alam dengan menggunakan input citra melalui integrasi image analysis. Dalam proses ini, teknik analisis seperti LBP (Local Binary Pattern) dan PCA (Principal Component Analysis) digunakan untuk mengolah pola tekstur lokal pada data citra bangunan. LBP menghasilkan fitur numerik yang kemudian direduksi dimensinya menggunakan PCA, sehingga pola-pola yang mendasari tingkat kerusakan dapat diidentifikasi dan dikelompokkan. Hasil PCA menunjukkan kemampuan yang signifikan dalam membedakan antara kerusakan ringan, sedang, dan berat berdasarkan pola dominan pada gambar bangunan pasca-bencana. Validasi oleh surveyor mengonfirmasi bahwa hasil pengolahan data ini sesuai dengan kondisi nyata di lapangan. Selain itu, sistem yang dikembangkan menunjukkan kekonsistenan pada data augmentasi seperti rotasi dan refleksi gambar, yang membuktikan keandalannya terhadap perubahan orientasi citra. Dengan demikian, sistem berbasis LBP dan PCA ini tidak hanya mempercepat proses penilaian kerusakan, tetapi juga memberikan dukungan objektif dan efisien bagi tim penilai dalam pengambilan keputusan mitigasi bencana.

ABSTRACT

Maghfira, Aulia Ilmi. 2024. ***Decision Support System Untuk Menentukan Tingkat Kerusakan Bangunan Menggunakan Metode PCA - Clustering***. Skripsi. Program Studi Teknik Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang. Pembimbing: (I) Dr. Agung Teguh Wibowo Almais, M.T (II) Ahmad Fahmi Karami, M.Kom

Damage to buildings due to natural disasters often requires rapid and accurate assessment to support mitigation measures. This research aims to determine the level of damage to buildings after natural disasters using image input through the integration of image analysis. In this process, analysis techniques such as LBP (Local Binary Pattern) and PCA (Principal Component Analysis) are used to process local texture patterns in building image data. LBP produces numerical features that are then reduced in dimension using PCA so that the patterns underlying the level of damage can be identified and categorized. The PCA results showed a significant ability to distinguish between light, medium, and heavy damage based on the dominant patterns in the post-disaster building images. Validation by surveyors confirmed that the results of this data processing match the real conditions on the ground. In addition, the developed system showed consistency in augmentation data such as image rotation and reflection, which proved its reliability against image orientation changes. Thus, this LBP and PCA-based system not only accelerates the damage assessment process but also provides objective and efficient support for the assessment team in disaster mitigation decision-making.

Key words: Building Damage, Natural Disasters, Local Binary Pattern (LBP), Principal Component Analysis (PCA), Image Analysis

الملخص

مغفرة، أولياء علمي. 2024. نظام دعم القرار لتحديد مستوى أضرار المباني باستخدام طريقة تحليل المكونات الرئيسية (PCA) للتصنيف. البحث الجامعي. قسم الهندسة المعلوماتية، كلية العلوم والتكنولوجيا بجامعة مولانا مالك إبراهيم الإسلامية الحكومية مالانج. المشرف الأول: د. أغونج تيغوه ويوويو أليس، الماجستير. المشرف الثاني: أحمد فهمي كرامي، الماجستير.

الكلمات الرئيسية: أضرار مباني، كوارث طبيعية، نمط ثنائي محلي، تحليل مكونات رئيسية (PCA)، تحليل صور.

غالبًا ما تتطلب أضرار المباني الناجمة عن الكوارث الطبيعية تقييمًا سريعًا ودقيقًا لدعم تدابير التخفيف. هدف هذا البحث إلى معرفة مستوى أضرار المباني بعد الكوارث الطبيعية باستخدام إدخال الصور من خلال تكامل تحليل الصور. في هذه العملية، يتم استخدام تقنية التحليل مثل LBP (النمط الثنائي المحلي) و PCA (تحليل المكونات الرئيسية) لمعالجة أنماط النسيج المحلية على بناء بيانات الصور. يولد LBP ميزات رقمية يتم تقليلها بعد ذلك حسب الأبعاد باستخدام PCA، بحيث يمكن تحديد الأنماط الأساسية لمستويات الضرر وتجميعها. أظهرت نتائج PCA قدرة كبيرة في التمييز بين الأضرار الخفيفة والمتوسطة والثقيلة بناءً على النمط السائد في صور المباني بعد الكارثة. وأكد التحقق من صحة المساح أن نتائج معالجة البيانات هذه كانت متوافقة مع الظروف الحقيقية في الميدان. بالإضافة إلى ذلك، أظهر النظام المطور اتساقًا في بيانات الزيادة مثل دوران الصورة وانعكاسها، مما أثبت موثوقيته مقابل تغيرات اتجاه الصورة. وبالتالي، فإن هذا النظام القائم على LBP و PCA لا يسرع عملية تقييم الأضرار فحسب، بل يوفر أيضًا دعمًا موضوعيًا وفعالًا لفريق التقييم في اتخاذ قرارات التخفيف من حدة الكوارث.

BAB I

PENDAHULUAN

1.1 Latar Belakang

Indonesia adalah negara yang terletak di kawasan Cincin Api Pasifik “*Ring Of Fire*” dimana merupakan salah satu wilayah dengan aktivitas tektonik yang aktif di dunia. Secara geografis Indonesia berada di pertemuan tiga lempeng tektonik utama, yaitu Lempeng Indo-Australia, Lempeng Eurasia, dan Lempeng Pasifik. Kondisi ini menyebabkan Indonesia menjadi salah satu negara yang rawan terhadap berbagai bencana alam seperti gempa bumi, letusan gunung berapi, dan tsunami. Dari data yang ada, gempa bumi di Indonesia rata-rata terjadi 5.000 kali dalam satu tahun, namun sejak tahun 2017 telah meningkat menjadi 7.000 kali dalam satu tahun, bahkan meningkat hingga 11.000 kali di tahun 2018 dan 2019 (Ardiyanti, 2023).

Kerusakan bangunan akibat bencana alam seperti gempa bumi, banjir, dan tanah longsor membutuhkan evaluasi yang cepat dan akurat agar langkah perbaikan dapat dilakukan secara tepat waktu. Pendekatan manual sering kali tidak memadai karena membutuhkan waktu yang lama dan rawan kesalahan dalam penilaian. Oleh karena itu, teknologi berbasis *Decision Support System* (DSS) sangat diperlukan untuk mempercepat proses penilaian kerusakan bangunan dan mendukung pengambilan keputusan yang lebih objektif (Sudipa *et al.*, 2023). Sistem ini dapat mengolah data lebih cepat dan mengurangi risiko kesalahan yang sering terjadi pada metode manual.

Decision Support System (DSS) sangat penting dalam membantu proses penilaian kerusakan bangunan pasca-bencana. DSS dapat memproses data kerusakan dengan sistematis, menghasilkan rekomendasi yang lebih objektif dan akurat. Sistem ini sangat relevan karena dapat membantu para profesional, seperti insinyur bangunan, dalam menilai kerusakan secara cepat dan efisien, terutama dalam kondisi darurat (Haryadi, 2009). DSS mampu menganalisis data kompleks waktu singkat dan membantu pemangku kepentingan membuat keputusan berbasis data.

Metode analisis seperti *Principal Component Analysis* (PCA) - *Clustering* memberikan efisiensi dalam pengolahan data yang kompleks. PCA berfungsi untuk mereduksi dimensi data yang besar dan kompleks, sehingga hanya informasi paling signifikan yang dipertahankan (Almais *et al.*, 2023). Dengan menghilangkan redundansi dalam data, DSS dapat memfasilitasi pengelompokan (*clustering*) untuk mengidentifikasi tingkat kerusakan bangunan yang berbeda. Proses ini meningkatkan akurasi penilaian dan mempercepat pengambilan keputusan terkait langkah mitigasi dan perbaikan.

Sebelum data diolah menggunakan PCA, diperlukan proses ekstraksi fitur untuk mendapatkan informasi penting dari citra bangunan yang rusak. Dalam hal ini, metode yang dipilih adalah *Local Binary Pattern* (LBP) karena kemampuannya dalam mengekstraksi pola tekstur dengan akurat dan efisien (Costaner *et al.*, 2024). LBP dipilih karena kesederhanaannya serta ketahanannya terhadap perubahan pencahayaan, yang sering kali tidak ideal dalam kondisi pasca-bencana. Metode ini menghasilkan deskriptor tekstur lokal yang relevan, sehingga informasi yang

diperlukan tetap dapat diperoleh meskipun kondisi pencahayaan kurang mendukung.

Setelah fitur diekstraksi menggunakan LBP, data tersebut diolah lebih lanjut dengan PCA untuk mereduksi dimensi data dan menyederhanakan kompleksitasnya. Proses ini memungkinkan analisis dan *clustering* yang lebih efisien, di mana bangunan yang mengalami kerusakan dapat dikelompokkan berdasarkan tingkat kerusakan yang dialami. Kombinasi antara LBP dan PCA ini menciptakan sistem yang lebih cepat dan akurat dalam menilai kerusakan bangunan, terutama dalam situasi kritis seperti pasca-bencana. Teknologi ini sangat membantu dalam mengidentifikasi kerusakan dengan lebih presisi dan mendukung upaya mitigasi yang lebih baik.

Dalam menghadapi bencana, manusia diajarkan untuk selalu berusaha melakukan tindakan preventif dan evaluatif. Sebagaimana disebutkan dalam QS Al-Hadid ayat 22:

مَا أَصَابَ مِنْ مُصِيبَةٍ فِي الْأَرْضِ وَلَا فِي أَنْفُسِكُمْ إِلَّا فِي كِتَابٍ مِّن قَبْلِ أَنْ نَبْرَأَهَا إِنَّ ذَلِكَ عَلَى اللَّهِ يَسِيرٌ ﴿٢٢﴾

"Tidak ada bencana (apa pun) yang menimpa di bumi dan tidak (juga yang menimpa) dirimu, kecuali telah tertulis dalam Kitab (Lauh Mahfuz) sebelum Kami mewujudkannya. Sesungguhnya hal itu mudah bagi Allah " (QS Al-Hadid: 22)

QS Al-Hadid ayat 22 menegaskan bahwa kehidupan dunia penuh dengan berbagai cobaan dan ujian dari Allah SWT, seperti rasa takut, kelaparan, kekurangan harta, jiwa, dan buah-buahan. Dalam "Tafsir Al-Mishbah", cobaan ini sebenarnya ringan jika dibandingkan dengan ganjaran besar yang dijanjikan bagi mereka yang bersabar. Ujian tersebut merupakan nikmat besar karena memberikan

kesempatan bagi manusia untuk mempersiapkan diri dan meningkatkan kualitas keimanan serta ketakwaan. Allah memerintahkan hamba-Nya untuk bersabar, mengingat segala sesuatu di dunia ini adalah milik-Nya, dan setiap cobaan pasti mengandung hikmah. Allah akan memberikan rahmat dan petunjuk kepada mereka yang sabar menghadapi ujian. Petunjuk ini akan membantu mereka mengatasi tantangan dan membawa kebahagiaan di dunia dan akhirat. Meskipun setiap bencana telah ditetapkan oleh Allah, manusia tetap dianjurkan untuk berusaha meminimalkan dampaknya. Pemanfaatan teknologi modern, seperti Sistem Pendukung Keputusan yang menggabungkan metode *Principal Component Analysis* (PCA) dan *Clustering*, merupakan bentuk ikhtiar manusia dalam menghadapi bencana. Dengan teknologi ini, proses pemulihan dapat dipercepat dan kerugian yang diakibatkan oleh bencana dapat diminimalkan. Usaha ini sejalan dengan amanah manusia untuk menjaga dan memperbaiki bumi setelah terjadi kerusakan.

Penerapan DSS berbasis PCA - *Clustering* memungkinkan proses penilaian kerusakan bangunan pasca-bencana dilakukan dengan lebih cepat, akurat, dan efisien. Teknologi ini menawarkan solusi praktis untuk kondisi darurat, memberikan hasil yang objektif serta mendukung pengambilan keputusan yang lebih tepat terkait mitigasi dan rekonstruksi. Kombinasi PCA untuk mereduksi dimensi data dan *Clustering* untuk pengelompokan tingkat kerusakan memberikan landasan yang kuat bagi sistem ini. Sistem ini sangat dapat diandalkan dalam situasi kritis di mana keputusan yang cepat dan akurat sangat diperlukan untuk mempercepat proses pemulihan dan rehabilitasi pasca-bencana.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah dibahas pada sub-bab sebelumnya, dapat dinyatakan permasalahan dalam penelitian ini sebagai berikut :

- a. Bagaimana menerapkan *Decision Support System* (DSS) untuk menentukan tingkat kerusakan bangunan pasca bencana alam berbasis *image*?
- b. Bagaimana menentukan tingkat kerusakan sektor pasca bencana alam menggunakan *Principal Component Analysis* (PCA)?

1.3 Batasan Masalah

Batasan masalah pada penelitian ini :

1. Pada penelitian ini menggunakan data citra kerusakan bangunan pasca bencana alam BPBD Kota Malang
2. Analisis hasil menggunakan perhitungan metode *Principal Component Analysis* (PCA)

1.4 Tujuan Penelitian

Berdasarkan rumusan masalah, tujuan dari penelitian ini adalah sebagai berikut :

1. Menerapkan *Decision Support System* (DSS) untuk menentukan tingkat kerusakan bangunan pasca bencana alam berbasis *image*
2. Menentukan tingkat kerusakan sektor pasca bencana alam menggunakan *Principal Component Analysis* (PCA)

1.5 Manfaat Penelitian

Dari uraian di atas, maka manfaat penelitian ini adalah sebagai berikut :

1. Meningkatkan efisiensi dan akurasi penilaian kerusakan bangunan serta mendukung pengambilan keputusan yang objektif yang dapat membantu dalam merencanakan dan melaksanakan tindakan pemulihan dan rekonstruksi pasca bencana dengan lebih efektif bagi surveyor.
2. Memberikan dasar bagi pengembangan DSS lebih lanjut di bidang teknologi informasi dan manajemen bencana khususnya pada penilaian kerusakan bangunan.

.

BAB II

STUDI PUSTAKA

2.1 Penelitian Tentang LBP-PCA

Terdapat penelitian terdahulu yang menunjukkan berbagai pendekatan yang dapat meningkatkan akurasi dan efektivitas evaluasi kerusakan dan rekonstruksi setelah bencana. Untuk menilai kerusakan dan rekonstruksi pasca bencana, implementasi produk Fuzzy-Weighted (FWP) dalam Decision Support System Dynamic (DSSD) adalah subjek penelitian yang dilakukan oleh (Almais & Fatchurrochman, 2020). Mereka menunjukkan bagaimana penggunaan FWP dapat meningkatkan akurasi dan efisiensi dalam penilaian kerusakan dengan mempertimbangkan berbagai variabel dan kondisi yang dinamis.

Sistem ini dibuat untuk membantu pengambilan keputusan dalam perencanaan dan pelaksanaan rekonstruksi dan rehabilitasi. Melalui pemanfaatan optimal data dari berbagai sumber, sistem dapat melakukan analisis mendalam dan prediktif terkait kerusakan yang terjadi. Sistem ini juga dilengkapi dengan kemampuan adaptif untuk menyesuaikan rekomendasi berdasarkan perubahan kondisi di lapangan. Integrasi data memberikan gambaran situasi yang komprehensif, sehingga keputusan dapat diambil secara informatif dan responsif. Hal ini membantu memastikan tindakan yang diambil sesuai prioritas kebutuhan lapangan, mempercepat proses bantuan, dan mengoptimalkan alokasi sumber daya secara efisien. Berikut ini adalah penelitian-penelitian terdahulu lainnya yang disajikan dalam tabel 2.1:

Tabel 2. 1 Penelitian Tentang LBP-PCA

No	Nama Peneliti	Topik	Metode	Hasil Penelitian
1.	Almais & Fatchurrochman, (Almais & Fatchurrochman, 2020)	<i>Fuzzy Weighted Product</i> untuk Rehabilitasi Pasca Bencana	Fuzzy-Weighted Product (FWP)	Implementasi DSS menggunakan FWP untuk penilaian kerusakan dan kerugian pascabencana.
2.	Safitri <i>et al.</i> , (2022)	<i>Optimization Query</i> Metode TOPSIS untuk Kerusakan Bencana	TOPSIS dengan optimasi query	Menguji optimasi query pada metode TOPSIS dalam DSS untuk meningkatkan efisiensi penilaian.
3.	Almais <i>et al.</i> , (2016)	<i>Multi-Criteria Decision Making</i> untuk Rehabilitasi	<i>Multi Experts Multi Criteria Decision Making</i> (MEMCDM)	MEMCDM digunakan untuk menyederhanakan proses penilaian tanpa perhitungan numerik yang rumit.
4.	Rahmaniati, (Rahmaniati, 2022)	TOPSIS Berbasis <i>Machine Learning</i> untuk Kerusakan Bencana	TOPSIS berbasis <i>Machine Learning</i>	Menggunakan TOPSIS dengan machine learning untuk penilaian kerusakan.
5.	Almais <i>et al.</i> , (2023)	<i>PCA-Based Clustering</i> untuk Kerusakan Pasca Bencana	<i>Principal Component Analysis</i> (PCA)	PCA digunakan untuk clustering data kerusakan dan memberi label tingkat kerusakan.
6.	Arhandi <i>et al.</i> , (2021)	<i>Local Binary Pattern</i> dan K-NN untuk Identifikasi Lubang Jalan	<i>Local Binary Pattern</i> (LBP) dan <i>K-Nearest Neighbor</i> (KNN)	LBP digunakan untuk mengidentifikasi pola tekstur pada citra jalan berlubang dan meningkatkan deteksi.
7.	Hayqal <i>et al.</i> , (2022)	K-Means dan PCA Dalam <i>Radial Basis Function Neural Network</i> (RBFNN) untuk Klasifikasi Data	PCA dan K-Means <i>Clustering</i> dalam <i>Radial Basis Function Neural Network</i> (RBFNN)	PCA untuk reduksi dimensi dan K-Means untuk inisialisasi center awal pada RBFNN, mendukung klasifikasi data.
8.	Alqoroni, (Alqoroni, 2023)	Implementasi COPRAS untuk Kerusakan Pasca Bencana	COPRAS	Implementasi COPRAS untuk menentukan tingkat kerusakan sektor pasca bencana dengan hasil klasifikasi yang tepat.
9.	Penelitian ini	Implementasi LBP dan PCA untuk Kerusakan Pasca Bencana	<i>Local Binary Pattern</i> dan <i>Principal Component Analysis</i>	Implementasi <i>Local Binary Pattern</i> (LBP) dan <i>Principal Component Analysis</i> (PCA) untuk menilai dan memberi label tingkat kerusakan pasca bencana alam dengan efisien.

Penelitian oleh Ichsan dkk. (2018) berfokus pada perbandingan algoritma PCA dan LBP untuk ekstraksi fitur dalam sistem pengenalan wajah. PCA digunakan untuk mereduksi dimensi data, mempertahankan informasi penting, dan mengurangi kompleksitas, sedangkan LBP digunakan untuk mengidentifikasi pola tekstur lokal pada citra wajah. Hasil penelitian menunjukkan bahwa LBP lebih efisien dalam kondisi pencahayaan yang bervariasi, dengan waktu pengenalan yang lebih cepat dan akurasi yang sama tinggi dibandingkan dengan PCA. Secara khusus, LBP mencapai akurasi 98,59% dengan waktu pengenalan 812,817 milidetik, sementara PCA membutuhkan 1275,761 milidetik untuk mencapai akurasi yang sama. Penelitian ini menyoroti bahwa LBP lebih efektif dalam skenario real-time dan kondisi pencahayaan yang berubah-ubah, meskipun PCA tetap unggul dalam mengurangi kompleksitas data.

2.2 Bencana Alam

Bencana alam adalah peristiwa yang disebabkan oleh proses alam yang mengakibatkan kerusakan lingkungan, kehilangan harta benda, dan korban jiwa. Beberapa jenis bencana alam yang sering terjadi di berbagai belahan dunia termasuk gempa bumi, letusan gunung berapi, banjir, angin topan, dan tanah longsor (Humairah *et al.*, 2023). Indonesia, sebagai negara kepulauan, memiliki potensi bencana alam yang tinggi, termasuk gelombang pasang, abrasi, dan tsunami (Ruswandi *et al.*, 2008).

Setiap jenis bencana alam memiliki karakteristik dan dampak yang berbeda. Misalnya, gempa bumi disebabkan oleh pergerakan lempeng tektonik dan dapat

mengakibatkan kerusakan struktural yang parah pada bangunan dan infrastruktur. Letusan gunung berapi terjadi akibat aktivitas vulkanik dan dapat menyemburkan abu serta lava yang merusak area sekitarnya. Banjir biasanya disebabkan oleh curah hujan yang tinggi atau luapan sungai, yang menggenangi daerah pemukiman dan pertanian.

Siklon tropis dan badai angin telah menyebabkan kerusakan yang luas, khususnya di Jawa dan Nusa Tenggara Timur (Yalesrie *et al.*, 2023). Tanah longsor terjadi ketika massa tanah dan batuan bergerak turun akibat gravitasi, sering kali dipicu oleh hujan lebat atau gempa bumi. Dampak dari bencana alam ini sangat signifikan, tidak hanya dari segi kerusakan fisik tetapi juga dari aspek sosial dan ekonomi. Oleh karena itu, penting untuk memiliki sistem mitigasi dan respons yang efektif untuk mengurangi kerugian dan mempercepat pemulihan pasca bencana.

2.3 Decision Support System (DSS)

Decision Support System (DSS) merupakan sistem informasi berbasis komputer yang membantu pengambilan keputusan dalam organisasi dengan menyediakan data, model, dan analisis (Rezty Amalia Aras, 2022). DSS dirancang untuk membantu pengambil keputusan dalam menangani masalah yang kompleks dan tidak terstruktur dengan menggunakan berbagai metode analitis dan komputasional.

Komponen inti DSS biasanya mencakup sistem manajemen basis data, sistem manajemen berbasis model, dan antarmuka pengguna atau sistem manajemen dialog (Wenzel, 2009). Basis data mengumpulkan dan mengelola data yang

diperlukan untuk analisis keputusan. Model base berisi model matematis dan statistik yang digunakan untuk menganalisis data dan menghasilkan berbagai alternatif keputusan.

DSS dapat diterapkan dalam berbagai bidang seperti manajemen bencana, pertanian, dan *real estate* untuk membantu pengambilan keputusan yang lebih efektif (Prihantara *et al.*, 2023). Dalam konteks penanggulangan bencana, DSS berfungsi untuk mengumpulkan data real-time dari berbagai sumber, menganalisis situasi bencana, dan memberikan rekomendasi tindakan yang harus diambil untuk mitigasi dan respons.

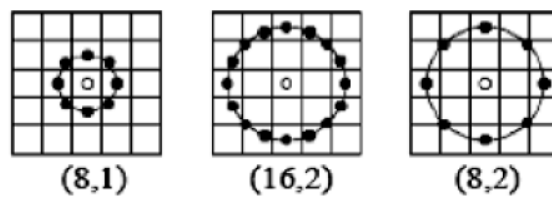
Misalnya, DSS dapat digunakan untuk memprediksi daerah yang berisiko tinggi terhadap bencana tertentu, merencanakan evakuasi, dan mengalokasikan sumber daya secara optimal. Dengan mengintegrasikan teknologi seperti analisis spasial, machine learning, dan simulasi, DSS dapat meningkatkan efektivitas pengambilan keputusan dan membantu mengurangi dampak negatif dari bencana. Implementasi sistem informasi untuk pengolahan data bencana dapat meningkatkan efisiensi dan efektivitas dalam pengambilan keputusan bagi lembaga penanggulangan bencana (Asmara *et al.*, 2017). Implementasi DSS yang efektif memerlukan data yang akurat, model yang tepat, dan antarmuka pengguna yang intuitif untuk memastikan bahwa pengambil keputusan dapat membuat keputusan yang cepat dan tepat berdasarkan informasi yang tersedia.

2.4 Local Binary Pattern (LBP)

LBP adalah metode ekstraksi ciri yang sangat efektif untuk mengidentifikasi pola tekstur pada citra *grayscale*. Metode ini diperkenalkan pertama kali oleh Ojala dkk. pada tahun 1994 dan telah menjadi salah satu teknik yang paling populer dalam analisis tekstur karena kesederhanaannya, kecepatan komputasinya, dan ketahanannya terhadap perubahan pencahayaan. LBP bekerja dengan cara membandingkan setiap piksel dalam citra dengan piksel-piksel tetangganya dan menghasilkan kode biner berdasarkan perbandingan tersebut.

Proses LBP melibatkan beberapa langkah utama sebagai berikut:

1. Pemilihan Piksel Tetangga:



Gambar 2. 1 Pola Ketetanggaan LBP
Sumber: Abdi *et al.*, (Abdi *et al.*, 2022)

LBP mengkaji nilai intensitas dari piksel pusat dan piksel-piksel tetangganya dalam suatu lingkungan tertentu (misalnya 3x3 piksel). Lingkungan ini biasanya berbentuk lingkaran dengan piksel pusat berada di tengah.

2. Pembentukan Kode Biner:

Nilai intensitas dari setiap piksel tetangga dibandingkan dengan nilai intensitas dari piksel pusat. Jika nilai intensitas tetangga lebih besar atau sama dengan nilai intensitas pusat, maka piksel tetangga tersebut diberi nilai 1, dan

jika tidak, diberi nilai 0. Hal ini menghasilkan delapan digit biner untuk setiap piksel pusat dalam lingkungan 3x3.

3. Penggabungan Kode Biner:

Digit-digit biner yang dihasilkan diatur menjadi satu nilai biner yang panjangnya 8 bit. Nilai ini kemudian dikonversi ke dalam bentuk desimal untuk setiap piksel pusat.

4. Pembuatan Histogram:

Nilai-nilai desimal yang dihasilkan dari seluruh piksel dalam citra digunakan untuk membentuk histogram yang mewakili distribusi tekstur dari citra tersebut.

Adapun rumus yang digunakan untuk menghitung nilai LBP pada lingkungan 3x3 dapat dilihat pada persamaan (2.1).

$$LBP(x, y) = \sum_{i=0}^{P-1} \frac{P-1}{i} s(g_i - g_c) \cdot 2^i \quad (2.1)$$

Keterangan:

g_i adalah nilai intensitas dari piksel tetangga ke- i ,

g_c adalah nilai intensitas dari piksel pusat,

Adapun fungsi threshold yang digunakan dalam *Local Binary Pattern* (LBP) dapat dilihat pada persamaan (2.2) berikut:

$$s(x) = \begin{cases} 1, & \text{jika } x \geq 0 \\ 0, & \text{jika } x < 0 \end{cases} \quad (2.2)$$

Adapun proses kerja *Local Binary Pattern* (LBP) pada lingkungan 3x3 dapat dilihat pada persamaan (2.3) berikut:

$$\begin{matrix} g_0 & g_1 & g_2 \\ g_7 & g_c & g_3 \\ g_6 & g_5 & g_4 \end{matrix} \quad (2.3)$$

LBP digunakan dalam berbagai aplikasi, terutama dalam pengenalan pola dan analisis tekstur. Kegunaan utama LBP meliputi pengenalan wajah, deteksi cacat, analisis citra medis, dan pemantauan kualitas material (Fahrurrozi *et al.*, 2023). Dalam analisis tekstur, LBP digunakan untuk mengekstraksi ciri-ciri tekstur dari berbagai objek, yang kemudian diaplikasikan dalam sistem pengenalan pola. Dalam deteksi cacat, LBP efektif dalam mendeteksi cacat pada permukaan material seperti kayu dan logam. LBP juga digunakan untuk menganalisis tekstur pada citra medis untuk mendeteksi kelainan atau penyakit. Selain itu, LBP dapat diterapkan dalam inspeksi kualitas pada berbagai industri untuk mendeteksi ketidaksempurnaan atau kerusakan pada produk.

LBP memiliki beberapa keunggulan, termasuk kesederhanaan komputasi, ketahanan terhadap perubahan pencahayaan, dan kemampuan untuk bekerja secara efektif dalam kondisi pencahayaan yang berbeda. Metode ini sangat berguna dalam aplikasi real-time di mana kecepatan dan akurasi sangat penting. Karena keefektifannya dalam mengidentifikasi pola tekstur pada citra digital, LBP menjadi pilihan ideal untuk analisis citra dalam berbagai kondisi dan aplikasi.

2.5 *Principal Component Analysis (PCA)*

PCA tidak hanya digunakan untuk mereduksi dimensi data, tetapi juga sangat efektif dalam meningkatkan performa metode clustering dengan menyederhanakan data berdimensi tinggi menjadi representasi yang lebih kompak dan informatif.

Dalam konteks ini, PCA membantu dalam mengidentifikasi pola dan struktur dalam data yang kompleks sehingga mempermudah proses pengelompokan (*clustering*) data (Wibowo Almais *et al.*, 2023). Berikut adalah penjelasan lebih rinci tentang proses PCA beserta langkah-langkah utama dan rumusnya:

1. Normalisasi Data

Normalisasi adalah langkah pertama dalam PCA. Normalisasi dilakukan untuk memastikan bahwa setiap variabel memiliki bobot yang sama dalam analisis. Tanpa normalisasi, variabel dengan skala yang lebih besar dapat mendominasi hasil analisis. Proses normalisasi mengubah skala data sehingga memiliki *Mean* 0 dan standar deviasi 1. Berikut adalah rumus normalisasi data standar yang digunakan dalam analisis, dapat dilihat pada persamaan (2.4) berikut:

$$x_{\text{standart}} = \frac{x - \text{mean}(x)}{\text{standart deviation}(x)} \quad (2.4)$$

Keterangan:

x_{standart}	= nilai yang dinormalisasi,
x	= nilai asli dari variabel,
$\text{Mean}(x)$	= rata-rata dari variabel
$\text{standart deviation}(x)$	= standar deviasi dari variabel

2. Perhitungan Matriks Kovariansi

Langkah kedua adalah menghitung matriks kovariansi. Matriks kovariansi digunakan untuk memahami hubungan antara berbagai variabel dalam data. Kovariansi mengukur sejauh mana dua variabel berubah bersama-sama. Matriks kovariansi yang digunakan dalam analisis ini dihitung menggunakan rumus yang dapat dilihat pada persamaan (2.5) berikut:

$$C = \frac{1}{n-1} \sum_{i=1}^n (x_i - \bar{x})^2 \quad (2.5)$$

Keterangan:

C = matriks kovariansi
 n = jumlah sampel
 x_i = vektor data pada titik i
 $\bar{x}$ = rata-rata vektor data

Matriks kovariansi memberikan gambaran mengenai bagaimana variabel-variabel dalam dataset saling berhubungan satu sama lain.

3. Dekomposisi Eigen

Setelah matriks kovariansi dihitung, langkah berikutnya adalah melakukan dekomposisi *eigen* pada matriks kovariansi. Dekomposisi *eigen* adalah proses untuk menemukan nilai *eigen*. Nilai *eigen* menunjukkan jumlah variansi yang ditangkap oleh setiap komponen utama, sedangkan vektor *eigen* menunjukkan arah dari komponen utama tersebut. Persamaan untuk dekomposisi *eigen* dalam analisis ini dapat dilihat pada persamaan (2.6) berikut:

$$Cv = \lambda v \quad (2.6)$$

Keterangan:

C = matriks kovariansi,
 v = vektor *eigen*,
 λ = nilai *eigen*

4. Pemilihan Komponen Utama

Langkah terakhir adalah pemilihan komponen utama. Vektor *eigen* diurutkan berdasarkan nilai *eigen* terbesar, dan komponen utama dipilih sesuai dengan jumlah variansi yang ingin dipertahankan. Rasio variansi yang

dijelaskan oleh setiap komponen utama diberikan oleh rumus pada persamaan (2.7) berikut:

$$variance\ ratio = \frac{\lambda_i}{\sum \lambda^j} \quad (2.7)$$

Keterangan:

λ_i = nilai *eigen* dari komponen utama ke-i,

$\sum \lambda^j$ = jumlah total dari semua nilai *eigen*.

Komponen utama yang pertama (PC1) biasanya menangkap sebagian besar variansi dalam data, diikuti oleh komponen utama kedua (PC2), dan seterusnya. Jumlah komponen utama yang dipilih bergantung pada seberapa banyak variansi yang ingin dipertahankan dalam analisis.

5. Penggunaan PCA

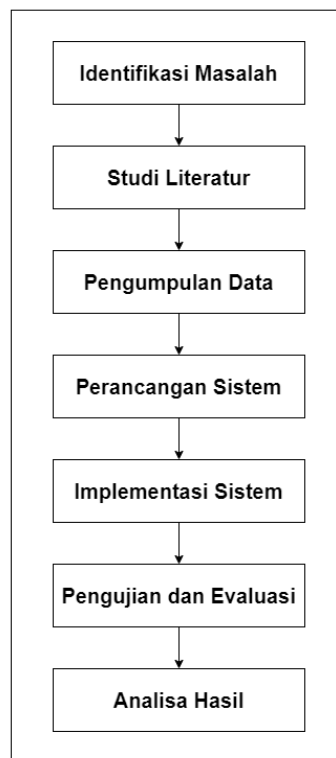
PCA sering digunakan dalam berbagai aplikasi seperti pengenalan pola, kompresi data, analisis gambar, dan sebagainya. Dalam konteks penelitian ini, PCA digunakan untuk melakukan clustering data kerusakan sektor pasca bencana dan memberi label tingkat kerusakan yang sesuai dengan kondisi di lapangan. PCA memungkinkan pengurangan kompleksitas data dan mempertahankan informasi signifikan, sehingga memudahkan dalam klasifikasi kerusakan bangunan dan mendukung pengambilan keputusan yang lebih baik dalam tindakan rehabilitasi dan rekonstruksi pasca bencana. PCA sangat bermanfaat karena dapat mengubah data yang kompleks dan berdimensi tinggi menjadi bentuk yang lebih sederhana, tanpa kehilangan informasi yang penting, sehingga analisis data menjadi lebih efisien dan efektif.

BAB III

METODE PENELITIAN

3.1 Desain Penelitian

Tujuan desain penelitian ini adalah untuk memberi penulis arahan yang jelas untuk mengikuti selama proses penelitian.



Gambar 3. 1 Desain Penelitian
Sumber: (Rana Puspita, 2023)

Langkah pertama adalah melakukan studi literatur untuk mengumpulkan referensi yang relevan, yang kemudian diikuti dengan identifikasi masalah untuk menganalisis kebutuhan penelitian. Selanjutnya, dilakukan pengumpulan data yang diperlukan untuk dasar perancangan sistem. Setelah data terkumpul, sistem dirancang dan diimplementasikan. Tahap berikutnya adalah pengujian dan evaluasi

untuk mengukur kinerja sistem serta akurasi hasil prediksi. Terakhir, hasil pengujian dan evaluasi dianalisis untuk interpretasi data dan pengambilan kesimpulan berdasarkan hasil yang diperoleh.

3.2 Pengumpulan Data

Data primer dan sekunder digunakan dalam penelitian ini. Data primer terdiri dari survei lapangan yang dilakukan oleh tim survei, foto yang diambil melalui kamera ponsel, dan pengumpulan data langsung yang dilakukan oleh peneliti. Data ini juga mencakup informasi yang diperoleh dari Badan Penanggulangan Bencana Daerah (BPBD) setempat. Dalam konteks ini, pengumpulan data primer sangat penting untuk memastikan keakuratan dan relevansi informasi yang digunakan dalam analisis. Untuk penelitian ini, data yang akan digunakan adalah data yang diperoleh dari BPBD Kota Malang. Data ini mencakup informasi terkini tentang kerusakan bangunan pasca bencana di wilayah Kota Malang, yang akan digunakan untuk analisis lebih lanjut.

Data sekunder adalah gambar yang diperoleh dari berbagai sumber online, termasuk situs web resmi pemerintah, jurnal ilmiah, laporan penelitian yang relevan, dan juga *Kaggle*. Data ini digunakan untuk melengkapi dan memverifikasi informasi yang diperoleh dari data primer. Penggunaan data sekunder juga membantu dalam memberikan konteks yang lebih luas terhadap temuan penelitian dan memastikan bahwa analisis yang dilakukan berbasis pada informasi yang komprehensif dan terkini. Data sekunder yang dikumpulkan mencakup informasi bencana dari situs web BPBD, laporan penelitian dari jurnal ilmiah yang terkait

dengan deteksi kerusakan bangunan dan analisis tekstur citra, serta data statistik dari situs web resmi pemerintah yang mencakup berbagai indikator terkait bencana alam.

Tabel 3. 1 Label Alternatif

Gambar	Label Alternatif
	Rusak Ringan
	Rusak Sedang
	Rusak Berat

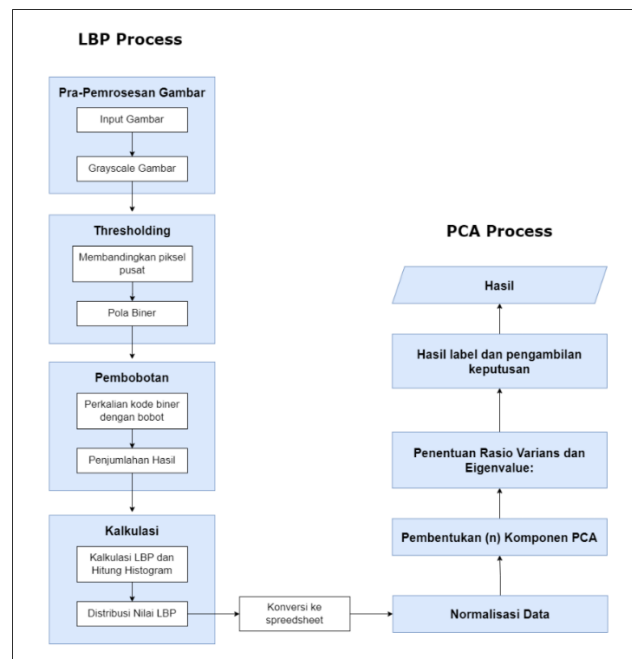
Sumber: (Fuadah, 2023)

Jumlah kelas yang ada pada data yang sudah didapatkan terdapat 3 kelas, yaitu rusak ringan, rusak sedang, dan rusak berat. Kelas-kelas ini akan digunakan untuk mengklasifikasikan tingkat kerusakan bangunan dalam penelitian ini. Label alternatif yang digunakan adalah sebagai berikut: Rusak Ringan, Rusak Sedang, dan Rusak Berat. Dengan menggabungkan data primer dan sekunder, penelitian ini dapat menghasilkan analisis yang lebih akurat dan menyeluruh mengenai kerusakan

bangunan pasca bencana alam. Data primer dari BPBD Kota Malang akan menjadi fokus utama dalam analisis, sementara data sekunder akan digunakan untuk mendukung dan memperkaya temuan penelitian.

3.3 Desain Sistem

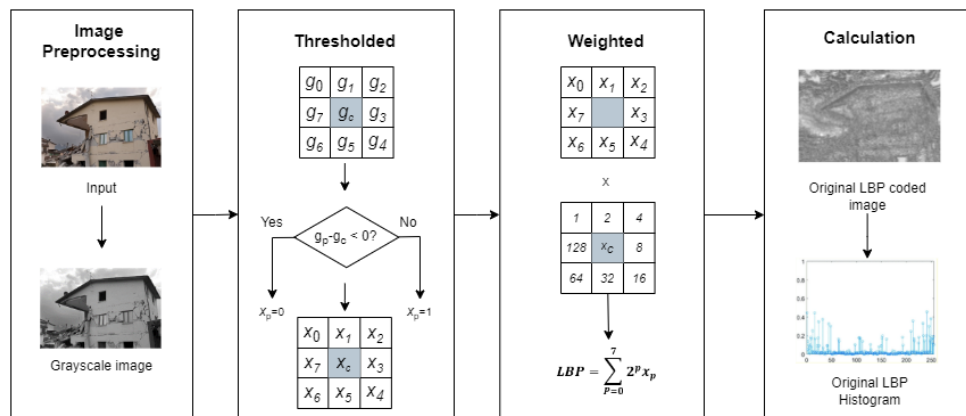
Bab ini memberikan penjelasan tentang proses yang digunakan untuk membuat sistem lebih mudah bagi peneliti untuk menerapkannya, melakukan evaluasi, dan menganalisis hasilnya. Gambar 3.2. menunjukkan desain sistem.. Sistem ini bertujuan untuk mendeteksi kerusakan bangunan pasca bencana alam dengan menggunakan metode *Local Binary Pattern* untuk ekstraksi fitur tekstur dan *Principal Component Analysis* untuk reduksi dimensi dan klasifikasi. Dengan menggabungkan kedua metode ini, sistem dapat memberikan hasil yang lebih akurat dan efisien dalam mengidentifikasi kerusakan bangunan.



Gambar 3. 2 Desain Sistem

Sumber: (He *et al.*, 2019), (Almais *et al.*, 2023), (Bovik, 2009).

Berikut adalah penjelasan detail tentang desain sistem pada gambar (3.2), yang menggambarkan alur proses dari implementasi metode *Local Binary Pattern* (LBP) dan *Principal Component Analysis* (PCA) dalam menentukan tingkat kerusakan bangunan:



Gambar 3. 3 Tahapan *Local Binary Pattern*
Sumber: He *et al.*, (2019)

Pada tahap Pra-Pemrosesan Gambar dalam metode LBP, gambar asli terlebih dahulu diubah menjadi gambar skala abu-abu untuk menyederhanakan analisis tekstur. Proses ini memastikan bahwa hanya satu kanal intensitas yang dianalisis, yang mengurangi kompleksitas dan memfokuskan pada pola intensitas tanpa memperhatikan warna. Setelah itu, tahap *Thresholded* membandingkan nilai piksel tetangga dengan piksel pusat untuk menghasilkan pola biner. Pola ini kemudian dikalikan dengan bobot pada tahap *Weighted* dan dijumlahkan untuk menghasilkan nilai LBP. Terakhir pada tahap kalkulasi, nilai-nilai LBP digunakan untuk membentuk citra baru dan histogram yang merepresentasikan distribusi tekstur dalam gambar. Berikut adalah tahapan-tahapan secara rinci menggunakan *Local Binary Pattern*.

1. Pemrosesan Gambar : Konversi Citra ke Skala Abu-abu



Gambar 3. 4 Proses *Grayscale*
Sumber: Aditya, (2021)

Konversi citra ke skala abu-abu merupakan langkah penting dalam pemrosesan citra, khususnya dalam ekstraksi fitur tekstur seperti LBP. Citra berwarna terdiri dari tiga saluran (merah, hijau, dan biru) yang masing-masing memiliki informasi intensitas yang berbeda. Proses ini bisa menambah kompleksitas perhitungan dan meningkatkan beban komputasi. Dengan mengonversi citra ke skala abu-abu, proses mereduksi dimensi data dari tiga saluran menjadi satu saluran tanpa kehilangan informasi penting yang berkaitan dengan pola intensitas lokal. Hal ini tidak hanya menyederhanakan perhitungan, tetapi juga memastikan konsistensi dan efisiensi dalam analisis tekstur, karena pola intensitas dalam skala abu-abu sudah cukup untuk mendeteksi variasi tekstur dan fitur yang diperlukan dalam metode LBP.

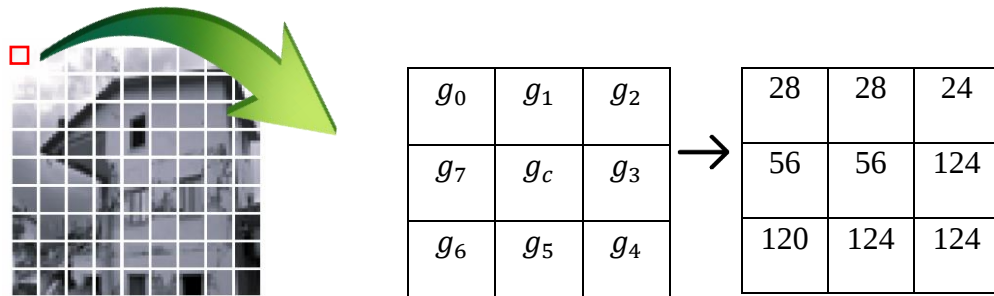
2. Thresholded

Pada tahap *thresholding* dalam metode LBP, gambar pertama-tama diubah ukurannya menjadi 81x81 piksel. Langkah ini dilakukan agar citra menjadi lebih seragam dan memudahkan dalam analisis tekstur. Setelah ukuran gambar disesuaikan, gambar tersebut diproses menggunakan blok-

blok 3x3 piksel yang diambil dari gambar yang telah di-*resize*. Setiap blok terdiri dari satu piksel pusat (g_c) dan delapan piksel tetangga (g_0 hingga g_7) di sekelilingnya.

$$\begin{bmatrix} 28 & 28 & 24 & \dots & 16 & 0 & 3 \\ 56 & 56 & 124 & \dots & 116 & 292 & 199 \\ 120 & 124 & 124 & \dots & 4 & 223 & 4 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 129 & 96 & 235 & \dots & 247 & 120 & 0 \\ 120 & 64 & 255 & \dots & 239 & 255 & 68 \\ 112 & 64 & 145 & \dots & 128 & 49 & 0 \end{bmatrix}$$

Langkah awal dalam penerapan LBP adalah mengonversi citra asli ke ukuran 81x81 piksel untuk memastikan analisis dilakukan pada ukuran yang seragam. Ukuran ini dipilih karena merupakan resolusi optimal untuk mempertahankan detail tekstur, sekaligus mempercepat proses perhitungan. Setelah gambar di-*resize*, setiap piksel pusat dalam gambar akan diproses dalam blok 3x3, yang akan digunakan dalam tahap *thresholding* untuk mendeteksi pola lokal. Matriks 3x3 dipilih karena efisien dalam mendeteksi pola-pola kecil dan perubahan intensitas di sekitar piksel pusat, yang kemudian digunakan untuk mendapatkan label baru yang merepresentasikan tekstur tersebut. Pendekatan ini efektif dalam menyederhanakan dan mengelompokkan pola tekstur, sehingga cocok untuk aplikasi seperti pendeteksian kerusakan bangunan (Achsani *et al.*, 2015).



Pada langkah *thresholding*, setiap piksel pusat (g_c) dalam blok 3x3 dibandingkan dengan delapan piksel tetangganya (g_0 hingga g_7). Setiap piksel tetangga di sekeliling piksel pusat akan dibandingkan secara individu dengan nilai intensitas dari piksel pusat. Jika nilai intensitas tetangga lebih besar atau sama dengan piksel pusat (g_c), maka tetangga tersebut diberi nilai biner 1. Jika intensitas tetangga lebih kecil dari piksel pusat, tetangga tersebut diberi nilai biner 0.

Analisis Multi-radius LBP

1) Radius 1 (P=8)

g_0	g_1	g_2
g_7	g_c	g_3
g_6	g_5	g_4

Menggunakan 8 titik tetangga dalam pola melingkar, dimana setiap titik dibandingkan dengan piksel pusat (g_c).

2) Radius 2 (P=16)

Menggunakan 16 titik tetangga dalam pola melingkar dengan radius 2 piksel, memberikan cakupan area yang lebih luas.

3) Radius 3, 4, dan 5 (P=24)

Menggunakan 24 titik tetangga untuk masing-masing radius, memungkinkan analisis tekstur pada skala yang lebih besar.

Hasil dari perbandingan ini menghasilkan pola biner 8-bit yang merepresentasikan tekstur lokal di sekitar piksel pusat.

Contoh perhitungan

28	28	24
56	56	124
120	124	124

Nilai piksel pusat (g_c) adalah 56. Setiap piksel tetangga dibandingkan dengan nilai ini:

Tabel 3. 2 Perbandingan Nilai Biner

Piksel Tetangga g_p	Nilai g_p	Perhitungan $g_p - g_c$	Hasil Perbandingan	Nilai Biner x_p
g_0	28	$10 - 15 = -5$	Kurang dari 0	$x_0 = 0$
g_1	28	$20 - 15 = 5$	Kurang dari 0	$x_1 = 0$
g_2	24	$10 - 15 = -5$	Kurang dari 0	$x_2 = 0$
g_3	124	$5 - 15 = -10$	Kurang dari 0	$x_3 = 1$
g_4	124	$10 - 15 = -5$	Tidak kurang dari 0	$x_4 = 1$
g_5	120	$25 - 15 = 10$	Tidak kurang dari 0	$x_5 = 1$
g_6	56	$10 - 15 = -5$	Tidak kurang dari 0	$x_6 = 1$
g_7	56	$10 - 15 = -5$	Tidak kurang dari 0	$x_7 = 1$

Pola biner yang dihasilkan adalah

28	28	24		0	0	0
56	56	124		1		1
120	124	124		1	1	1

→

0	0	0
1		1
1	1	1

Pola ini disusun menjadi urutan 8-bit: 0001111

Setelah histogram dihitung dan dinormalisasi, lima fitur statistik utama diekstraksi untuk menggambarkan karakteristik distribusi histogram: *Mean*, *Variance*, *Entropy*, *Energy*, dan *Skewness*. *Mean* menggambarkan rata-rata distribusi, sedangkan *Variance* menunjukkan tingkat penyebaran nilai histogram. *Entropy* mengukur tingkat ketidakaturan distribusi histogram, sementara *Energy* menggambarkan kekuatan total distribusi. *Skewness* digunakan untuk mengukur asimetri distribusi histogram. Kelima fitur ini memberikan informasi rinci mengenai tekstur dalam tiap grid, yang relevan untuk langkah analisis lebih lanjut.

1. *Mean* menunjukkan rata-rata nilai histogram di setiap grid.
2. *Variance* menggambarkan distribusi nilai histogram di sekitar rata-rata.
3. *Entropy* mencerminkan tingkat kerandoman pola tekstur, di mana nilai tinggi menunjukkan lebih banyak keragaman.
4. *Energy* mewakili konsistensi pola, di mana nilai tinggi menunjukkan pola yang lebih homogen.

Skewness mengindikasikan asimetri distribusi histogram, yang memberikan informasi tambahan tentang pola tekstur. Setelah mendapatkan nilai LBP, citra dibagi menjadi grid 5x5. Untuk setiap sel grid, dihitung fitur statistik berikut:

1) *Mean* (μ)

Mean dihitung untuk menentukan nilai rata-rata dari piksel dalam grid, menggunakan persamaan (3.1):

$$\mu = (1/N) * \sum x_i \quad (3.1)$$

Keterangan:

N = jumlah piksel dalam grid

x_i = nilai LBP piksel ke-i

2) *Variance* (σ^2)

Variance mengukur dispersi nilai LBP dari Mean, menunjukkan variabilitas tekstur, menggunakan persamaan (3.2):

$$\sigma^2 = (1/N) * \sum (x_i - \mu)^2 \quad (3.2)$$

Mengukur dispersi nilai LBP dari *Mean*, menunjukkan variabilitas tekstur.

3) *Entropy* (H)

Entropy digunakan untuk mengevaluasi kompleksitas tekstur, menggunakan persamaan (3.3):

$$H = -\sum p(i) * \log_2(p(i)) \quad (3.3)$$

Dimana $p(i)$ adalah probabilitas kemunculan nilai LBP i . *Entropy* tinggi menunjukkan tekstur yang kompleks.

4) *Energy* (E)

Energy menggambarkan homogenitas tekstur, dihitung menggunakan persamaan (3.4):

$$E = \sum p(i)^2 \quad (3.4)$$

Energy tinggi menunjukkan tekstur yang homogen.

5) *Skewness* (S)

Skewness mengevaluasi distribusi nilai piksel terhadap Mean, menggunakan persamaan (3.5):

$$S = (1/N) * \sum ((x_i - \mu)/\sigma)^3 \quad (3.5)$$

Fitur-fitur statistik ini memberikan karakterisasi kuantitatif dari pola tekstur lokal pada berbagai skala, yang kemudian dapat digunakan untuk mengklasifikasikan tingkat kerusakan bangunan. Kombinasi multi-radius dengan analisis statistik grid memungkinkan deteksi variasi tekstur yang lebih komprehensif dan robust.

3. *Weighted*

Tahap *Weighted* dalam metode LBP adalah proses di mana pola biner yang telah dihasilkan pada tahap *thresholding* dikonversi menjadi nilai desimal yang unik. Ini dilakukan dengan mengalikan setiap bit dalam pola biner dengan bobot tertentu, yang tergantung pada posisi bit tersebut dalam lingkungan 3x3. Bobot ini berbentuk pangkat dari dua, sehingga setiap bit biner dikalikan dengan bobot yang sesuai, dan hasilnya dijumlahkan untuk mendapatkan nilai LBP.

Pola biner, setelah tahap *thresholding*, menghasilkan pola biner 8-bit yang dihasilkan dari perbandingan piksel pusat dengan piksel-piksel tetangganya.

Dimana $x_0, x_1, \dots, x_7$ adalah nilai biner (0 atau 1) yang diperoleh dari hasil *thresholding*. Pemberian bobot pada setiap bit dalam pola biner diberi bobot sesuai dengan posisinya dalam lingkungan 3x3. Bobot ini adalah pangkat dari dua:

x_0	x_1	x_2	=	1	2	4
x_7	x_c	x_3		128	x_c	8
x_6	x_5	x_4		64	32	16

Perhitungan nilai LBP dihitung dengan menjumlahkan hasil perkalian bit biner dengan bobotnya masing-masing. Rumus untuk menghitung nilai LBP adalah:

$$LBP = \sum_{p=0}^7 2^p \cdot X_p \quad (3.6)$$

Setiap bit X_p (yang bisa 0 atau 1) dikalikan dengan bobot 2^p yang sesuai dengan posisinya, lalu semua hasil perkalian dijumlahkan untuk mendapatkan nilai LBP. Dalam hal ini, piksel pusat (g_c) tidak dihitung. Setiap bit dari pola biner dikalikan dengan bobot yang sesuai dengan posisinya.

Pola biner 00011111 di tahap *thresholding* menghasilkan bit sebagai berikut:

Tabel 3. 3 Perhitungan Nilai Biner

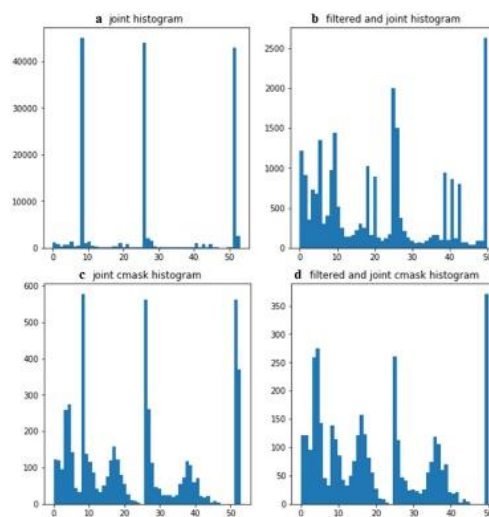
Bit	Bobot	Perhitungan Bit * Bobot
$x_0 = 0$	1	$0 * 1 = 0$
$x_1 = 0$	2	$0 * 2 = 0$
$x_2 = 0$	4	$0 * 4 = 0$
$x_3 = 1$	8	$1 * 8 = 8$
$x_4 = 1$	16	$1 * 16 = 16$
$x_5 = 1$	32	$1 * 32 = 32$
$x_6 = 1$	64	$1 * 64 = 64$
$x_7 = 1$	128	$1 * 128 = 128$

Langkah selanjutnya adalah menjumlahkan semua hasil perkalian untuk mendapatkan nilai LBP:

$$\begin{aligned}
& 0*1 + 0*2 + 0*4 + 1*8 + 1*16 + 1*32 + 1*64 + 1*128 \\
& = 0 + 0 + 0 + 8 + 16 + 32 + 64 + 128 \\
& = 248
\end{aligned}$$

Hasil perhitungan nilai LBP di atas menunjukkan bahwa setiap piksel tetangga dibandingkan dengan piksel pusat untuk menentukan nilai biner (0 atau 1) berdasarkan intensitasnya. Nilai biner ini kemudian dikalikan dengan bobot yang sesuai berdasarkan posisi tetangganya, dan semua hasil perkalian dijumlahkan untuk menghasilkan nilai LBP akhir. Dalam contoh ini, nilai LBP yang diperoleh adalah 248. Proses ini memberikan representasi numerik dari pola tekstur lokal di sekitar piksel pusat, yang nantinya bisa digunakan sebagai fitur dalam analisis citra untuk mengidentifikasi atau mengklasifikasikan pola tertentu.

4. Kalkulasi



Gambar 3. 5 Histogram LBP
Sumber: Le *et al.*, (2020)

Setelah menghitung nilai LBP untuk setiap piksel dalam sebuah gambar berukuran 81x81 akan mendapatkan sebuah matriks LBP 81x81 yang besar.

Matriks ini merepresentasikan tekstur gambar pada resolusi yang lebih tinggi. Pada setiap piksel dalam gambar 81x81, metode *Local Binary Pattern* (LBP) diterapkan melalui langkah *thresholding* dan pembobotan (*Weighted*). Setelah semua piksel diproses, matriks LBP 81x81 ditampilkan sebagai hasil langsung dari perhitungan LBP. Matriks ini merepresentasikan distribusi tekstur dengan resolusi tinggi. Setiap elemen dalam matriks adalah hasil perhitungan LBP untuk masing-masing piksel dalam gambar asli.

$$\begin{bmatrix} 28 & 28 & 24 & \dots & 16 & 0 & 3 \\ 56 & 56 & 124 & \dots & 116 & 292 & 199 \\ 120 & 124 & 124 & \dots & 4 & 223 & 4 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 129 & 96 & 235 & \dots & 247 & 120 & 0 \\ 120 & 64 & 255 & \dots & 239 & 255 & 68 \\ 112 & 64 & 145 & \dots & 128 & 49 & 0 \end{bmatrix}$$

Setelah proses downsampling pertama, nilai rata-rata dari setiap blok 3x3 dihitung menggunakan persamaan (3.7), di mana k adalah jumlah elemen dalam blok (dalam hal ini k=9).. Dengan demikian, meskipun data dalam matriks disederhanakan, informasi tekstur utama yang diperlukan untuk analisis tetap terjaga.

$$\text{Rata-rata Blok} = \frac{1}{k} \sum_{i=1}^k \text{nilai LBP} \quad (3.7)$$

Dimana k adalah jumlah elemen dalam blok (pada kasus ini k=9 untuk blok 3x3). Matriks ini tetap mempertahankan informasi tekstur penting, namun dalam format yang lebih sederhana dibandingkan matriks 81x81.

$$\begin{bmatrix} 248 & 104 & 171 & \dots & 98 & 134 & 85 \\ 91 & 118 & 207 & \dots & 174 & 196 & 101 \\ 93 & 172 & 116 & \dots & 157 & 118 & 145 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 144 & 137 & 85 & \dots & 154 & 110 & 172 \\ 141 & 141 & 130 & \dots & 142 & 173 & 99 \\ 134 & 70 & 167 & \dots & 185 & 168 & 122 \end{bmatrix}$$

Untuk seluruh gambar 81×81, dihitung LBP untuk setiap piksel, menghasilkan matriks LBP. Matriks ini kemudian dibagi menjadi grid 5×5 untuk analisis statistik.

Fitur statistik yang dihitung untuk setiap grid:

1. $Mean (\mu) = \Sigma(hist_norm[i] \times i)$
2. $Variance (\sigma^2) = \Sigma((i - \mu)^2 \times hist_norm[i])$
3. $Entropy (H) = -\Sigma(hist_norm[i] \times \log_2(hist_norm[i] + \epsilon))$
4. $Energy (E) = \Sigma(hist_norm[i]^2)$
5. $Skewness (S) = \Sigma((i - \mu)^3 \times hist_norm[i])/\sigma^3$

Hasil akhir adalah matriks fitur dengan dimensi 25×5, dimana setiap baris merepresentasikan satu sel grid dan setiap kolom merepresentasikan fitur statistik.

5. Data Asli PCA

Data yang digunakan adalah matriks fitur 25×5 dari hasil ekstraksi LBP, dimana setiap baris adalah satu sel grid dan kolom merepresentasikan fitur statistik (*Mean, Variance, Entropy, Energy, Skewness*).

6. Menghitung Deviasi Standar (Standard Deviation)

Deviasi standar adalah ukuran seberapa jauh data tersebar dari rata-ratanya. Langkah ini penting untuk menormalkan data lebih lanjut sebelum menghitung kovarians.

Untuk menghitung rata-rata dan standar deviasi untuk kolom pertama (sebagai contoh berikut) :

Rata-rata (μ) untuk kolom pertama :

$$\mu = \frac{128+178+194+208+205+187+112+121+128}{9} = 162.34$$

Rumus untuk menghitung deviasi standar (σ) dapat dilihat pada persamaan (3.8) berikut:

$$\sigma_j = \sqrt{\frac{1}{N} \sum_{i=1}^n (x_{ij} - \mu_j)^2} \quad (3.8)$$

Keterangan:

σ_j = deviasi standar dari kolom j

n = jumlah sampel

Standar deviasi (σ) untuk kolom pertama : $\sigma = 37.06151$

Standarisasi untuk elemen pertama dari kolom pertama :

$$Z_1 = \frac{128 - 162.34}{37.06151} = -0.92639$$

Proses ini diulangi untuk semua elemen dan semua kolom, menghasilkan matriks yang distandarisasi:

Data yang distandarisasi =

$$\begin{bmatrix} -0.9264 & -1.3793 & \cdots & -1.0996 & -0.5412 \\ 0.4227 & 0.5622 & \cdots & -1.5362 & 2.1806 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ -1.1153 & -1.2322 & \cdots & 0.4528 & -0.0398 \\ -0.9264 & -0.4379 & \cdots & 1.3746 & 0.3899 \end{bmatrix}$$

7. Menghitung Matriks Kovarians (Covariance Matrix)

Matriks kovarians mengukur bagaimana dua variabel dalam dataset berubah bersama-sama. Dalam PCA, matriks kovarians digunakan untuk menemukan arah utama variasi dalam data.

Rumus untuk menghitung kovarians antara dua variabel X dan Y dapat dilihat pada persamaan (3.9) berikut:

$$Cov(X, Y) = \frac{1}{n-1} \sum_{i=1}^n (X_i - \mu_X)(Y_i - \mu_Y) \quad (3.9)$$

Keterangan:

X dan Y = dua fitur dari dataset.
 μ_X dan μ_Y = rata-rata dari fitur X dan Y .
 n = jumlah data.

Setiap elemen dari matriks kovarians dihitung dengan menggunakan rumus di atas untuk setiap pasangan kolom dari data yang distandarisasi:

Matriks Kovarians =

$$\begin{bmatrix} 1.000 & 0.325 & 0.421 & -0.512 & 0.234 \\ 0.325 & 1.000 & 0.287 & -0.326 & 0.412 \\ 0.421 & 0.287 & 1.000 & -0.445 & 0.365 \\ -0.512 & -0.326 & -0.445 & 1.000 & -0.287 \\ 0.234 & 0.412 & 0.365 & -0.287 & 1.000 \end{bmatrix}$$

8. Menghitung *Eigenvalues* dan *Eigenvectors*

Eigenvalues dan *eigenvectors* dari matriks kovarians dihitung untuk menentukan arah (*eigenvectors*) dan besarnya (*eigenvalues*) variasi dalam data. *Eigenvectors* menentukan arah utama variabilitas, sementara *eigenvalues* menentukan besarnya variasi sepanjang arah tersebut.

Rumus dasar untuk menemukan *eigenvalues* dan *eigenvectors* dapat dilihat pada persamaan (3.10) berikut:

$$A \cdot v = \lambda \cdot v \quad (3.10)$$

Keterangan:

A = matriks kovarians.
 v = *eigenvector*
 λ = *eigenvalue*

Eigenvalues mewakili besarnya variansi yang dijelaskan oleh masing-masing komponen utama, di mana semakin besar nilai *eigenvalue*, semakin banyak informasi atau variasi yang dijelaskan oleh komponen tersebut. Sementara itu, *eigenvectors* menunjukkan arah di mana variansi data paling besar, dengan setiap *eigenvector* terkait langsung dengan satu *eigenvalue*. Kombinasi keduanya memberikan gambaran mengenai dimensi utama data yang paling signifikan.

9. Proyeksi Data ke Ruang Komponen Utama

Setelah menghitung *eigenvalues* dan *eigenvectors*, data asli diproyeksikan ke ruang komponen utama untuk mengurangi dimensi dataset sambil mempertahankan sebanyak mungkin variasi asli. Biasanya, hanya beberapa komponen utama pertama yang dipilih, karena mereka menjelaskan sebagian besar variabilitas dalam data.

Proses proyeksi data dilakukan dengan mengalikan data yang telah dinormalisasi dengan matriks *eigenvectors*, seperti yang ditunjukkan pada persamaan (3.11) berikut:

$$Projected\ Data = Z \times V \quad (3.11)$$

Keterangan:

Z = data yang telah dinormalisasi.

V = matriks *eigenvector* yang telah diurutkan.

10. Visualisasi

Pada metode ini, PCA digunakan untuk mereduksi dimensi data kerusakan bangunan yang telah diolah dengan LBP. Visualisasi hasil dari PCA sangat penting untuk membantu memahami pola variansi dalam data serta mendukung proses pengambilan keputusan. Dengan menampilkan data kerusakan bangunan dalam visualisasi tersebut, analisis terhadap tingkat kerusakan menjadi lebih mudah dan efektif, karena data yang kompleks disederhanakan tanpa kehilangan informasi signifikan.

Secara teknis, proses visualisasi dimulai dengan menormalkan data hasil ekstraksi LBP. Setelah itu, PCA diterapkan untuk mereduksi dimensi dengan mempertahankan variansi terbesar dalam data, dengan setiap titik merepresentasikan citra yang telah dianalisis dan dikelompokkan berdasarkan tingkat kerusakan. Pada penelitian (Almais *et al.*, 2024), labeling atau pemberian label dilakukan dengan membagi data ke dalam kategori, seperti kerusakan ringan, sedang, atau berat, yang kemudian divisualisasikan berdasarkan komponen utama PC1. Proses ini memudahkan dalam melihat pola dan kluster tingkat kerusakan secara lebih jelas dan informatif.

BAB IV

HASIL DAN PEMBAHASAN

4.1 Uji Coba Sistem

Dalam uji coba sistem ini, metode *Local Binary Pattern* (LBP) dan *Principal Component Analysis* (PCA) diterapkan untuk menganalisis data citra kerusakan bangunan yang diperoleh dari Badan Penanggulangan Bencana Daerah (BPBD) Kota Malang. Sebanyak 12 gambar digunakan sebagai sampel untuk mengidentifikasi dan mengelompokkan tingkat kerusakan bangunan. LBP digunakan untuk mengekstraksi fitur tekstur dari citra, sedangkan PCA dimanfaatkan untuk mereduksi dimensi data guna mempermudah proses pengelompokan dan meningkatkan efisiensi komputasi.

Hasil uji coba menunjukkan bahwa sistem mampu mengelompokkan tingkat kerusakan bangunan ke dalam tiga kategori, yaitu rusak ringan, rusak sedang, dan rusak berat. Kategori ini ditentukan berdasarkan nilai Principal Component yang telah dinormalisasi, di mana rentang nilai tertentu merepresentasikan masing-masing tingkat kerusakan. Validasi eksternal dilakukan dengan mencocokkan hasil pengelompokan sistem terhadap data referensi yang telah diverifikasi oleh ahli di bidang penilaian kerusakan bangunan atau surveyor lapangan.

4.2 Hasil Uji Coba

4.2.1 Proses LBP

A. Pra-pemrosesan Gambar

Pra-pemrosesan gambar merupakan tahap awal yang sangat penting dalam analisis citra digital. Proses ini bertujuan untuk mempersiapkan data gambar agar lebih mudah dianalisis dan diolah dengan metode yang dirancang, seperti *Local Binary Pattern* (LBP) dan *Principal Component Analysis* (PCA). Pada penelitian ini, gambar yang digunakan berasal dari data bencana alam BPBD Kota Malang.



Gambar 4. 1 Gambar RGB

Tahap pertama dalam pra-pemrosesan adalah mengambil gambar input dalam format RGB. Gambar-gambar ini merupakan dokumentasi visual dari kerusakan bangunan yang terjadi di wilayah Malang, yang diperoleh dari arsip BPBD Kota Malang. Format RGB memungkinkan penyimpanan informasi warna dari setiap gambar, yang penting untuk menangkap detail visual awal sebelum proses lebih lanjut. Namun, karena tujuan analisis adalah mendeteksi pola tekstur dan intensitas, informasi warna ini perlu disederhanakan. Oleh karena itu, gambar input RGB ini menjadi dasar untuk tahapan berikutnya, yaitu konversi ke *grayscale*.



Gambar 4. 2 Gambar *Grayscale*

Setelah memperoleh gambar dalam format RGB, langkah selanjutnya adalah mengonversi gambar tersebut ke format *grayscale*. Konversi ini dilakukan untuk menyederhanakan data dan mengurangi dimensi warna menjadi hanya satu kanal intensitas. Dalam gambar *grayscale*, setiap piksel hanya memiliki nilai intensitas yang merepresentasikan tingkat kecerahan, dari hitam (intensitas rendah) hingga putih (intensitas tinggi). Proses ini tidak hanya mengurangi kompleksitas data, tetapi juga mempertahankan informasi tekstur penting yang dibutuhkan dalam analisis pola kerusakan. Dengan format *grayscale*, pola-pola seperti retakan atau deformasi bangunan dapat lebih mudah diidentifikasi.



Gambar 4. 3 Gambar *resize*

Tahap terakhir dalam pra-pemrosesan adalah mengubah ukuran gambar *grayscale* menjadi resolusi 81x81 piksel. Proses *resize* ini dilakukan untuk

menyamakan dimensi seluruh gambar, sehingga data dapat diolah secara konsisten. Resolusi 81x81 dipilih untuk menjaga keseimbangan antara efisiensi komputasi dan kelengkapan informasi pola tekstur. Dengan ukuran ini, pola kerusakan tetap terlihat jelas, namun jumlah data yang diproses menjadi lebih manageable untuk metode clustering seperti PCA. Hasil akhir dari proses *resize* ini adalah gambar yang siap untuk diekstraksi fitur dan dianalisis lebih lanjut menggunakan algoritma yang telah dirancang. Berikut adalah *pseudocode* untuk *pra-preprocessing* data :

```
Function ExtractLBPFeatures(image_path):
    Input: image_path (string, path to the image file)
    Output: image_resized_81x81 (processed grayscale image)

    Step 1: Read the image in RGB format
        img_rgb ← ReadImage(image_path)
        If img_rgb is None Then
            RaiseError("File not found: " + image_path)
        End If

    Step 2: Display the original RGB image
        DisplayImage(ConvertToRGB(img_rgb), Title="Original
Image (RGB)")

    Step 3: Convert the RGB image to Grayscale
        img_gray ← ConvertToGrayscale(img_rgb)
        DisplayImage(img_gray, Title="Grayscale Image")

    Step 4: Resize the Grayscale image to 81x81 pixels
```

```

        image_resized_81x81 ← ResizeImage(img_gray, Width=81,
Height=81)

        DisplayImage(image_resized_81x81, Title="Resized Image
(81x81)")

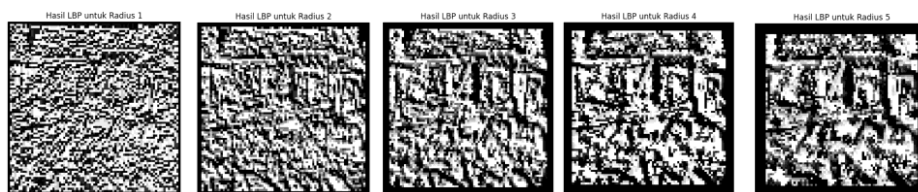
        Step 5: Return the resized grayscale image

        Return image_resized_81x81
    End Function

```

B. Thresholding

Thresholding merupakan tahap kunci dalam metode *Local Binary Pattern* (LBP) yang digunakan untuk menganalisis tekstur dalam gambar. Tahap ini bertujuan untuk mengekstrak pola-pola lokal dari gambar dengan membandingkan nilai intensitas piksel pusat terhadap piksel-piksel tetangganya. Hasil dari proses ini adalah pola biner yang dapat diubah menjadi nilai numerik unik untuk setiap piksel dalam gambar. Proses ini memberikan representasi tekstur lokal yang menjadi dasar untuk analisis lebih lanjut.



Gambar 4. 4 Pola Radius

Dalam proses *thresholding*, setiap piksel pada gambar berfungsi sebagai pusat, sementara piksel-piksel tetangganya diidentifikasi berdasarkan radius tertentu. Nilai intensitas piksel pusat dibandingkan dengan nilai intensitas piksel-

piksel tetangganya. Jika nilai intensitas tetangga lebih besar atau sama dengan nilai intensitas pusat, maka piksel tetangga diberi nilai biner 1. Sebaliknya, jika nilai intensitas tetangga lebih kecil, maka nilai biner yang diberikan adalah 0. Pola biner yang dihasilkan kemudian diubah menjadi nilai desimal menggunakan metode pengkodean biner. Nilai desimal ini dikenal sebagai nilai LBP untuk piksel tersebut.

Hasil *thresholding* menghasilkan representasi tekstur yang kaya, di mana setiap nilai LBP menggambarkan pola lokal di sekitar piksel pusat. Proses ini dilakukan secara menyeluruh untuk setiap piksel pada gambar, sehingga menghasilkan citra LBP yang mencerminkan pola-pola tekstur yang ada dalam gambar.

Dengan kombinasi parameter seperti (8,1), (8,2), (16,2), dan (24,3), hasil optimal diperoleh pada parameter (P, R) (24,3), yang menunjukkan bahwa tekstur lokal dengan radius yang lebih luas dan titik yang lebih banyak mampu menangkap pola tekstur yang lebih kompleks dan signifikan. Hal ini menunjukkan bahwa parameter seperti radius 3 dengan 24 titik memiliki kemampuan yang lebih baik dalam merepresentasikan pola tekstur dari data (Kosasih & Daomara, 2021). Dalam konteks analisis kerusakan bangunan, penggunaan parameter serupa memungkinkan sistem untuk menangkap pola tekstur dari berbagai tingkat kerusakan, memberikan representasi yang akurat dari kerusakan ringan hingga berat sesuai dengan variasi pola global yang diamati.

Maka dari itu, dibuatlah analisis menggunakan hingga 5 radius dengan berbagai jumlah titik untuk menangkap pola tekstur secara lebih menyeluruh. Setiap radius memiliki cakupan dan detail yang berbeda, mulai dari radius 1 dengan 8 titik

yang fokus pada pola lokal kecil, hingga radius 5 dengan 24 titik yang mencerminkan pola global. Hasil dari proses *thresholding* divisualisasikan melalui penerapan LBP pada radius yang berbeda, mulai dari radius 1 hingga radius 5. Setiap radius mencerminkan area pengambilan sampel yang semakin luas:

1. Radius 1 dengan 8 Points

Pada radius ini, tetangga-tetangga yang dianalisis berada di sekitar piksel pusat dengan jarak 1 piksel, membentuk lingkaran kecil yang terdiri dari 8 titik. Titik-titik tersebut diposisikan secara merata di sekeliling piksel pusat. Dengan radius kecil ini, LBP menangkap detail tekstur lokal yang sangat kecil, seperti tepi atau sudut tajam di sekitar piksel pusat. Analisis ini sangat berguna untuk pola-pola lokal berfrekuensi tinggi.

2. Radius 2 dengan 16 Points

Jarak tetangga dari piksel pusat diperbesar menjadi 2 piksel, dengan 16 titik yang diambil secara merata di sekitar piksel pusat. Cakupan area yang lebih luas ini memungkinkan LBP menganalisis pola-pola tekstur yang lebih besar dibandingkan radius 1. Ini berguna untuk menangkap detail tekstur yang sedikit lebih kompleks dan memperluas informasi lingkungan.

3. Radius 3 dengan 24 Points

Radius ini memiliki tetangga yang berjarak 3 piksel dari piksel pusat, dengan 24 titik yang membentuk lingkaran lebih besar. Pola tekstur yang dianalisis pada radius ini mencerminkan transisi dari pola lokal menuju pola global. Analisis ini cocok untuk menangkap area dengan perubahan tekstur yang lebih halus, seperti gradient atau struktur yang lebih besar dalam gambar.

4. Radius 4 dengan 24 Points

Dengan radius 4, tetangga berada pada jarak 4 piksel dari piksel pusat. Tetap menggunakan 24 titik, pola tekstur yang dihasilkan mulai menunjukkan pola global yang lebih dominan, meskipun tetap mempertahankan beberapa detail lokal. Radius ini cocok untuk area dengan struktur tekstur yang lebih besar dan lebih homogen, seperti permukaan datar dengan variasi kecil.

5. Radius 5 dengan 24 Points

Radius terbesar dalam analisis ini adalah 5 piksel, dengan 24 titik yang membentuk lingkaran luas di sekitar piksel pusat. Cakupan area yang sangat besar ini menghasilkan pola tekstur yang mencerminkan pola global kasar dalam gambar. Representasi pada radius ini berguna untuk menangkap fitur besar yang tersebar di gambar, seperti bentuk objek utama atau distribusi pola dominan.

Thresholding dalam LBP memungkinkan ekstraksi informasi tekstur yang rinci dari sebuah gambar. Dengan membandingkan nilai piksel pusat dengan tetangganya, metode ini mampu menangkap pola-pola yang bervariasi dari detail kecil hingga pola yang lebih besar. Visualisasi hasil *thresholding* menunjukkan kemampuan LBP dalam mendeteksi pola tekstur pada berbagai skala radius, sehingga memberikan gambaran tekstur lokal hingga global dari gambar. Berikut adalah *pseudocode*:

```
Function CalculateLBP(image, radius, n_points):  
    Input:  
        image (2D array, grayscale image)
```

```

    radius (int, distance from center pixel)

    n_points (int, number of neighboring points)

Output:

    lbp_image (2D array, LBP-transformed image)

Step 1: Initialize the LBP output image

    height ← GetHeight(image)

    width ← GetWidth(image)

    lbp_image ← InitializeArray(height, width, Value=0)

Step 2: Iterate through each pixel in the image

    For i from radius to (height - radius):
        For j from radius to (width - radius):
            center ← image[i, j]

            binary_pattern ← EmptyList()

            Step 2.1: Calculate binary pattern for neighbors
            For k from 0 to (n_points - 1):
                theta ← 2 * π * k / n_points # Angle for
neighbor
                y ← int(i + radius * sin(theta)) # Y-
coordinate of neighbor
                x ← int(j + radius * cos(theta)) # X-
coordinate of neighbor

                If  $0 \leq y < \text{height}$  AND  $0 \leq x < \text{width}$  Then

```

```

                                AppendToList(binary_pattern, 1 if
image[y, x] ≥ center Else 0)

                                Else
                                AppendToList(binary_pattern, 0)
                                End If
                                End For

Step 2.2: Compute LBP value for the pixel
lbp_value ← 0
For idx, val in Enumerate(binary_pattern):
    lbp_value ← lbp_value + (val * (2idx))
End For

lbp_image[i, j] ← lbp_value
End For
End For

Step 3: Return the LBP-transformed image
Return lbp_image
End Function

```

C. Pembobotan

Pembobotan dilakukan dengan memberikan bobot pada setiap nilai biner dari pola biner yang dihasilkan pada proses *thresholding*. Setiap bit dalam pola biner diberikan bobot berdasarkan posisinya, menggunakan pangkat dua (2^{idx}). Nilai

akhir diperoleh dengan menjumlahkan hasil perkalian setiap bit dengan bobotnya.

Proses ini dapat direpresentasikan melalui potongan kode berikut:

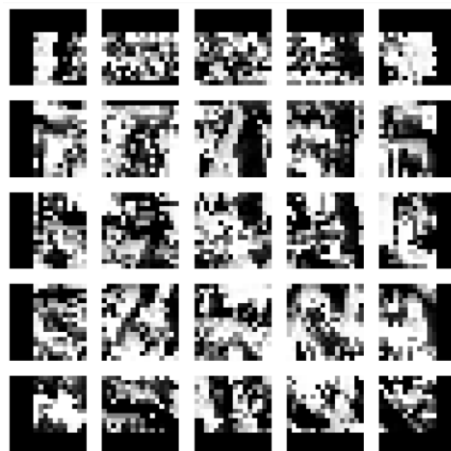
```
# Pembobotan dilakukan dalam fungsi calculate_lbp:

lbp_value = sum([val * (2 ** idx) for idx, val in
enumerate(binary_pattern)])

lbp_image[i, j] = lbp_value
```

D. Kalkulasi

Tahap awal pada proses kalkulasi adalah membagi gambar hasil LBP menjadi grid kecil berukuran 5x5. Pembagian ini dilakukan untuk memecah gambar menjadi area lokal yang lebih kecil, sehingga pola-pola detail yang ada pada setiap bagian dapat dianalisis secara mendalam. Dimensi grid ditentukan dengan membagi tinggi dan lebar gambar hasil *resize* (81x81) menjadi lima bagian yang sama besar. Dengan cara ini, setiap grid mewakili bagian spesifik dari gambar dan memberikan formasi lokal yang signifikan terkait pola LBP. Grid ini menjadi dasar untuk menghitung histogram distribusi nilai LBP pada tiap bagian gambar.



Gambar 4. 5 Grid 5x5

Setelah gambar terbagi menjadi grid, nilai-nilai LBP dari berbagai radius (1 hingga 5) pada grid tersebut dikombinasikan. Kombinasi ini dilakukan untuk menangkap pola-pola tekstur dari berbagai skala, mulai dari skala kecil hingga besar. Nilai LBP untuk masing-masing radius dihimpun menjadi satu array menggunakan fungsi `np.hstack`, yang menghasilkan satu representasi nilai LBP untuk tiap grid. Kombinasi ini memastikan bahwa semua pola tekstur pada radius tertentu dapat dimanfaatkan secara maksimal dalam analisis statistik berikutnya.

Histogram normalisasi digunakan untuk menganalisis distribusi nilai LBP pada setiap grid. Histogram ini dihitung dengan membagi nilai LBP ke dalam 64 bins dengan rentang nilai 0-256, yang mencerminkan tingkat intensitas pola pada grid tersebut. Selanjutnya, histogram dinormalisasi sehingga semua nilainya berada dalam rentang $[0, 1]$. Normalisasi ini memastikan bahwa histogram dari grid dengan jumlah nilai yang berbeda dapat dibandingkan secara objektif. Proses ini memberikan gambaran jelas mengenai pola distribusi tekstur pada setiap grid secara lokal.

Tabel 4. 1 Hasil Ekstraksi

Index	Mean	Variance	Entropy	Energy	Skewness
0	10.853627	413.713031	2.444578	0.431381	7.625915
1	14.400362	563.345145	2.715925	0.349474	7.385682
2	13.382246	541.047728	2.588861	0.367084	7.440193
3	13.066298	526.212916	2.488501	0.381053	7.451928
4	12.585756	484.739739	2.548657	0.395890	7.554731
5	15.920495	604.769297	2.660509	0.349005	7.386814
6	25.433048	738.650076	3.578454	0.143919	5.198853
7	23.337989	659.838278	3.866103	0.109844	4.403624
8	25.213450	678.817012	3.831657	0.108085	4.383581
9	16.293886	591.394894	3.016422	0.270855	7.114540
10	14.708260	559.380617	2.831325	0.333267	7.385244
11	24.892351	688.334021	3.936388	0.108363	4.740857
12	25.943284	711.749022	3.668552	0.124651	4.831663
13	25.409704	675.077426	3.980088	0.105136	4.551344

Index	Mean	Variance	Entropy	Energy	Skewness
14	18.243408	630.476250	3.048353	0.260602	7.014576
15	14.237132	531.684577	2.806631	0.333187	7.391184
16	25.767568	728.054083	3.756434	0.128619	5.200779
17	27.424779	721.082100	3.828125	0.112730	4.343447
18	30.725857	732.946652	3.650198	0.119147	4.161584
19	23.436364	770.891405	2.845187	0.264008	6.651157
20	12.392216	490.082694	2.516129	0.396106	7.517455
21	18.250965	629.226592	3.245570	0.245069	7.076251
22	15.677298	537.321755	3.241265	0.239992	7.035170
23	19.031189	641.308969	3.174017	0.249524	7.007173
24	14.888158	573.204597	2.578998	0.368729	7.435657

Hasil perhitungan fitur statistik dari setiap grid disimpan ke dalam sebuah list, yang kemudian diubah menjadi DataFrame menggunakan library pandas. Setiap baris DataFrame merepresentasikan satu grid dengan lima kolom fitur statistik yang telah dihitung. Berdasarkan data yang diberikan, hasil klasifikasi "rusak berat" kemungkinan besar dihasilkan oleh pola karakteristik fitur yang memiliki nilai-nilai signifikan, seperti *Mean*, *Variance*, *Entropy*, *Energy*, dan *Skewness*, yang secara kolektif menunjukkan adanya kerusakan yang parah. Berikut adalah penjelasannya:

- 1) *Mean* yang Tinggi: Pada beberapa titik (contoh: indeks 6, 7, 8, 11, 12, 13, 16, 17, 18), nilai *Mean* melebihi 23. Nilai rata-rata yang tinggi ini menunjukkan bahwa intensitas piksel dalam tekstur gambar memiliki variasi besar, yang sering ditemukan pada pola kerusakan berat.
- 2) *Variance* yang Tinggi: *Variance*, yang menggambarkan penyebaran intensitas piksel, sangat tinggi di titik-titik seperti indeks 6–13, menunjukkan pola tekstur yang lebih tidak seragam dan beragam. Pola ini umum terjadi pada kerusakan berat, di mana kerusakan melibatkan area yang lebih luas dan pola tekstur yang kacau.

- 3) *Entropy* Tinggi: *Entropy* yang tinggi pada beberapa indeks (contoh: 6, 7, 8, 11, 12, 13) menunjukkan bahwa distribusi intensitas piksel pada gambar sangat kompleks dan tidak teratur. Kerusakan berat sering menghasilkan pola tekstur yang acak, yang mencerminkan tingkat *Entropy* yang tinggi.
- 4) *Energy* yang Rendah: *Energy* adalah indikator kekuatan intensitas piksel yang terdistribusi. Pada beberapa indeks yang dianggap "rusak berat" (contoh: 6–13), nilai *Energy* rendah (sekitar 0.1), menunjukkan distribusi yang tersebar dan kurang terpusat, yang konsisten dengan pola kerusakan berat.
- 5) *Skewness* Rendah: Nilai *Skewness* rendah (contoh: indeks 6–13 dengan nilai sekitar 4–5) menunjukkan bahwa distribusi intensitas piksel tidak memiliki kecenderungan tertentu. Hal ini terjadi pada pola kerusakan berat yang memiliki banyak elemen tekstur yang tidak beraturan.

Indeks-indeks dengan karakteristik fitur seperti *Mean*, *Variance*, dan *Entropy* yang tinggi, serta *Energy* dan *Skewness* yang rendah, memberikan kontribusi signifikan pada variansi yang dijelaskan oleh PC1. PC1, sebagai komponen utama, memprioritaskan pola yang menunjukkan tekstur lebih tidak beraturan, kontras tinggi, dan kompleksitas tinggi—semua indikator kerusakan berat.

Dengan representasi ini, data tekstur gambar dapat dianalisis secara efisien atau digunakan untuk langkah berikutnya, seperti reduksi dimensi dengan PCA atau clustering. DataFrame ini menjadi keluaran akhir tahap kalkulasi dan berfungsi sebagai dasar untuk menyederhanakan data serta mengambil kesimpulan terkait karakteristik gambar. Berikut adalah *pseudocode*:

```
Function ExtractFeaturesFromGrid(image, lbp_images, grid_size):
```

```

Input:

    image (2D array, resized image)

    lbp_images (Dictionary, LBP-transformed images with keys
as radii)

    grid_size (int, size of the grid to divide the image)

Output:

    features_list (List of dictionaries containing
statistical features)

Initialize variables:

    grid_height ← GetHeight(image) / grid_size
    grid_width ← GetWidth(image) / grid_size
    features_list ← EmptyList()

For i from 0 to (grid_size - 1):
    For j from 0 to (grid_size - 1):
        grid_combined ← EmptyList()

        For radius in [1, 2, 3, 4, 5]:
            grid_values ← Flatten(
                lbp_images[radius][
                    i * grid_height : (i + 1) * grid_height,
                    j * grid_width : (j + 1) * grid_width
                ]
            )
            AppendToList(grid_combined, grid_values)
        End For
    End For

```

```

        hist ← ComputeHistogram(grid_combined, Bins=64,
Range=(0, 256))

        If Sum(hist) ≠ 0 Then

            hist ← NormalizeHistogram(hist)

        Else

            hist ← InitializeArray(64, Value=0)

        End If

        Mean_lbp ← ComputeMean(hist)
        Variance_lbp ← ComputeVariance(hist, Mean_lbp)
        Entropy_lbp ← ComputeEntropy(hist)
        Energy_lbp ← ComputeEnergy(hist)
        Skewness_lbp ← ComputeSkewness(hist)

        features ← {
            "Mean": Mean_lbp,
            "Variance": Variance_lbp,
            "Entropy": Entropy_lbp,
            "Energy": Energy_lbp,
            "Skewness": Skewness_lbp
        }

        AppendToList(features_list, features)

    End For

End For

Return features_list

End Function

```

4.2.2 Proses PCA

A. Normalisasi Data

Normalisasi dilakukan dengan mengubah setiap nilai fitur menjadi nilai relatif terhadap rata-rata dan distribusi data dalam kolom tersebut. Setelah proses ini, setiap kolom akan memiliki rata-rata nol dan standar deviasi satu. Hal ini memastikan bahwa setiap fitur memiliki kontribusi yang setara dalam analisis, tanpa bias terhadap fitur tertentu yang memiliki rentang nilai lebih besar. Misalnya, fitur seperti *Variance* yang memiliki rentang nilai besar dapat menjadi setara dalam bobot analisis dengan fitur seperti *Skewness* yang memiliki rentang nilai lebih kecil.

```
[[-1.4668852 -2.21418672 -1.34613843 1.60380714 0.99729171]
 [-0.84803742 -0.58387507 -0.83116411 0.87219023 0.81219205]
 [-1.02568214 -0.82681582 -1.07231192 1.02949235 0.85419244]
 [-1.08080994 -0.98844802 -1.26278043 1.15426331 0.86323491]
 [-1.16465682 -1.44031762 -1.14861387 1.28679143 0.94244423]
 [-0.58279899 -0.13253962 -0.93633616 0.86800897 0.81306386]
 [ 1.07698729 1.32615389 0.80578132 -0.96387729 -0.87276631]
 [ 0.71143338 0.4674626 1.35169304 -1.26825035 -1.48549281]
 [ 1.03867098 0.67424476 1.28631979 -1.28396311 -1.50093611]
 [-0.51764833 -0.27825998 -0.26086776 0.16994715 0.60327621]
 [-0.79431426 -0.62707045 -0.6121539 0.72743247 0.81185403]
 [ 0.98264441 0.77793701 1.4850838 -1.28147618 -1.22565409]
 [ 1.16601505 1.03305436 0.97677361 -1.13598362 -1.15568756]
 [ 1.07291398 0.63350023 1.56801979 -1.31030183 -1.37167407]
 [-0.17748836 0.14754961 -0.20026806 0.07836274 0.5262535 ]
 [-0.8765184 -0.92883172 -0.65901917 0.7267131 0.81643079]
 [ 1.13535545 1.21070561 1.14355968 -1.10053981 -0.87128226]
 [ 1.42451192 1.1347426 1.27961777 -1.24247202 -1.53185982]
 [ 2.00049646 1.26401243 0.94193896 -1.18515242 -1.67198563]
 [ 0.72859819 1.6774382 -0.58584623 0.10879047 0.24623839]
 [-1.19842645 -1.38210363 -1.21034638 1.28872368 0.91372371]
 [-0.17616969 0.13393401 0.17401903 -0.06037673 0.57377425]
 [-0.62523281 -0.86741204 0.16584842 -0.10573215 0.54212161]
 [-0.04003329 0.26557714 0.03822166 -0.02058634 0.5205493 ]
 [-0.76292501 -0.47645175 -1.09103047 1.0441888 0.85069767]]
```

Gambar 4. 6 Data Normalisasi

Hasil pada tabel menunjukkan data yang telah dinormalisasi menggunakan metode `StandardScaler`. Setiap baris merepresentasikan satu pengamatan, sedangkan kolom menunjukkan fitur yang telah dihitung, seperti *Mean*, *Variance*, *Entropy*, *Energy*, dan *Skewness* dari setiap grid LBP yang sebelumnya diekstraksi. Nilai-nilai ini dinormalisasi sehingga memiliki rata-rata nol dan standar deviasi satu, memungkinkan fitur-fitur dengan skala berbeda untuk dianalisis secara setara. Sebagai contoh, nilai pada baris pertama (-1.4668852 hingga 0.99729171) menggambarkan sebaran data yang mencakup berbagai karakteristik statistik dari grid LBP. Data ini menjadi input untuk analisis lebih lanjut seperti PCA, yang membutuhkan data ter-standarisasi untuk menentukan komponen utama tanpa bias akibat skala fitur yang berbeda. Normalisasi ini memastikan setiap fitur memiliki kontribusi yang seimbang dalam proses analisis.

```
Function NormalizeAndCategorizeFeatures(principalDF, min_val,
max_val):
    Input:
        principalDF (DataFrame, contains principal component
data)
        min_val (float, minimum value for normalization range)
        max_val (float, maximum value for normalization range)
    Output:
        principalDF (DataFrame with added normalized and
categorized columns)

    Step 1: Normalize PC1
    pc1_min ← MinimumValue(principalDF['PC1'])
```

```

    pc1_max ← MaximumValue(principalDF['PC1'])

    For each row in principalDF:
        row['PC1'] ← min_val + ((row['PC1'] - pc1_min) * (max_val
- min_val)) / (pc1_max - pc1_min)
    End For

    Step 2: Categorize based on normalized PC1 values
    categories ← EmptyList()

    For each row in principalDF:
        pc1_value ← row['PC1']

        If pc1_value < 0 Then
            AppendToList(categories, "Rusak Ringan")
        Else If 0 ≤ pc1_value < 2 Then
            AppendToList(categories, "Rusak Sedang")
        Else If pc1_value ≥ 2 Then
            AppendToList(categories, "Rusak Berat")
        End If
    End For

    Step 3: Add categorized data to DataFrame
    principalDF['Kategori'] ← categories

    Return principalDF
End Function

```

Proses normalisasi menggunakan StandardScaler adalah langkah penting dalam analisis data, terutama saat data memiliki berbagai fitur dengan rentang nilai

yang berbeda. Dalam konteks ini, normalisasi diterapkan pada dataset fitur yang mencakup nilai seperti *Mean*, *Variance*, *Entropy*, *Energy*, dan *Skewness* yang dihasilkan dari proses ekstraksi histogram LBP. Tujuannya adalah untuk menyamakan skala semua fitur sehingga tidak ada satu fitur pun yang mendominasi analisis berikutnya, khususnya pada langkah PCA.

B. Pembentukan Komponen PCA

Komponen utama (Principal Components) dihitung berdasarkan nilai *eigenvector* dari matriks kovariansi data. *Eigenvector* ini mewakili arah dari variabilitas data yang maksimum, sedangkan *eigenvalue* menentukan seberapa besar variasi data yang dapat dijelaskan oleh masing-masing komponen. Komponen pertama (PC1) memiliki variansi terbesar, diikuti oleh komponen kedua (PC2), dan seterusnya. Dalam eksperimen ini, tiga komponen utama diambil karena dianggap cukup untuk merepresentasikan informasi utama dari data tanpa kehilangan terlalu banyak detail.

Hasil dari pembentukan komponen PCA ditampilkan dalam bentuk tabel yang menunjukkan nilai PC1 untuk setiap pengamatan. Nilai-nilai ini merupakan representasi baru dari data dalam ruang PCA, yang lebih sederhana namun tetap mencakup informasi penting. Setiap nilai di dalam tabel merupakan hasil proyeksi dari data asli ke arah komponen utama yang telah dihitung.

Tabel 4. 2 PC 1

0	1	2	3	4
3.390785	1.771826	2.154593	2.394599	2.668172
5	6	7	8	9
1.506276	-2.246522	-2.377576	-2.599275	0.819562
10	11	12	13	14
1.600407	-2.581744	-2.446831	2.783706	0.380970
15	16	17	18	19
1.787556	-2.345544	-2.961295	-2.615589	0.674884
20	21	22	23	24
-1.577177	2.770798	-3.113410	-0.615589	1.900584

Tabel 4.2 menyajikan nilai PC1 yang diperoleh dari hasil analisis Principal Component Analysis (PCA). Setiap baris dan kolom pada tabel ini merepresentasikan distribusi nilai PC1 dari data citra pascabencana. Nilai-nilai ini menunjukkan kontribusi variabel terhadap komponen utama pertama (PC1) yang merepresentasikan dimensi terbesar variansi dalam data. Nilai-nilai ini mencerminkan posisi data pada dimensi utama yang memuat informasi variansi terbesar setelah proses reduksi dimensi dengan PCA. Nilai positif seperti 3.390785 pada indeks 0 menunjukkan bahwa data tersebut berada pada rentang yang lebih jauh di sisi positif PC1, mengindikasikan pola kerusakan yang lebih signifikan. Sebaliknya, nilai negatif seperti -3.113410 pada indeks 22 mengindikasikan pola kerusakan yang relatif rendah dalam ruang komponen utama. Distribusi nilai-nilai PC1 ini digunakan untuk klasifikasi kategori kerusakan, seperti "Rusak Ringan," "Rusak Sedang," atau "Rusak Berat," sesuai dengan kriteria berbasis nilai ambang yang telah ditentukan. Data ini merupakan hasil akhir dari analisis tekstur dengan LBP dan reduksi dimensi menggunakan PCA, yang mempermudah visualisasi dan pengambilan keputusan terkait tingkat kerusakan. Berikut adalah pseudo codenya:

```
Function PerformPCA(data, num_components):
```

```

Input:
    data (DataFrame or matrix, contains normalized feature
data)
    num_components (integer, number of principal components
to retain)
Output:
    principalDF (DataFrame containing the principal
components)

Step 1: Initialize PCA model
pca_model ← InitializePCA(n_components = num_components)

Step 2: Fit and transform the data
principal_components ← TransformDataWithPCA(pca_model, data)

Step 3: Create a DataFrame for principal components
principalDF ← CreateDataFrame(principal_components,
columns=['PC1', 'PC2', ..., 'PC<num_components>'])

Return principalDF
End Function

```

C. Penentuan rasio varians dan *eigenvalue*

Pada tahap ini dilakukan analisis untuk menentukan seberapa besar kontribusi masing-masing komponen utama (PC1, PC2, dan PC3) dalam menjelaskan variasi total dari data awal.

Tabel 4. 3 Explained Variance Ration

PC	Explained Variance Ration Untuk Setiap Komponen :
PC 1	91.82%
PC 2	6.07%
PC 3	1.81%

Berdasarkan hasil analisis PCA, dapat disimpulkan bahwa komponen utama pertama (PC1) memiliki kontribusi sebesar 91.82% terhadap total variasi data, menjadikannya komponen yang paling signifikan dalam menjelaskan variasi data dibandingkan dengan PC2 dan PC3. Dikarenakan PC2 hanya berkontribusi sebesar 6.07% dan PC3 sebesar 1.81%, maka PC1 dianggap cukup untuk merepresentasikan mayoritas informasi yang terdapat dalam data. Oleh karena itu, keputusan untuk menggunakan PC1 sebagai komponen utama dalam analisis data dianggap paling tepat tanpa perlu melibatkan PC2 dan PC3 yang kontribusinya relatif kecil.

Rasio varians ini diperoleh dari perhitungan *explained_Variance_ratio* pada model PCA, yang mengindikasikan persentase informasi yang dapat dijelaskan oleh setiap komponen utama. Informasi ini penting untuk memutuskan berapa banyak komponen utama yang perlu digunakan dalam analisis selanjutnya, tergantung pada seberapa besar variasi data yang ingin dipertahankan.

Tabel 4. 4 Eigen Value

PC	Eigen Value Untuk Setiap Komponen :
PC 1	4.78
PC 2	0.32
PC 3	0.09

Eigenvalue adalah representasi numerik dari varians yang dijelaskan oleh masing-masing komponen utama. Pemilihan PC1 sebagai komponen utama didasarkan pada kemampuannya untuk menjelaskan sebagian besar variasi dalam data, dengan *eigenvalue* sebesar 4.78, jauh lebih tinggi dibandingkan PC2 (0.32)

dan PC3 (0.09). Nilai ini menunjukkan bahwa PC1 merepresentasikan informasi paling signifikan dalam data, memungkinkan efisiensi informasi tanpa kehilangan detail penting. Selain itu, penggunaan PC1 memberikan reduksi dimensi yang optimal, sehingga analisis menjadi lebih sederhana dan efisien tanpa menambah kompleksitas yang tidak perlu dari komponen dengan kontribusi kecil seperti PC2 dan PC3.

Nilai *eigen* ini dihitung dari matriks kovarian data setelah normalisasi. Komponen utama dengan *eigenvalue* yang lebih besar menunjukkan bahwa komponen tersebut mampu menjelaskan lebih banyak informasi atau variasi data dibandingkan dengan komponen lainnya.

```
Function ExtractKeyPCASStatistics(pca_model):

    Input:

        pca_model (PCA object after fitting data)

    Output:

        highest_Variance_ratio (float): Highest explained
Variance ratio

        largest_eigenvalue (float): Largest eigenvalue

    Step 1: Extract the highest explained Variance ratio
    highest_Variance_ratio =
max(pca_model.explained_Variance_ratio_)

    Step 2: Extract the largest eigenvalue
    largest_eigenvalue = max(pca_model.explained_Variance_)
```

```

Return highest_Variance_ratio, largest_eigenvalue

End Function

```

D. Hasil Label dan Pengambilan Keputusan

Pada tahap ini, dilakukan proses penentuan label kategori berdasarkan nilai komponen utama (PC1) yang telah dinormalisasi. Proses ini bertujuan untuk mengklasifikasikan data ke dalam kategori tertentu berdasarkan tingkat kerusakan yang diidentifikasi dari hasil PCA.

Nilai PC yang merupakan hasil dari *Principal Component Analysis* (PCA) dinormalisasi agar berada dalam rentang nilai tertentu, yaitu antara -2 hingga 4. Proses normalisasi ini bertujuan untuk menyelaraskan skala data sehingga memudahkan dalam proses pembagian kategori. Rumus normalisasi yang digunakan adalah transformasi linear berdasarkan nilai minimum dan maksimum PC. Setelah nilai PC dinormalisasi, dilakukan evaluasi terhadap setiap data dalam DataFrame principalDF. Setiap nilai PC dibandingkan dengan rentang nilai yang telah ditentukan, sebagaimana yang ditampilkan dalam tabel berikut:

Tabel 4. 5 Kategori Kerusakan Bangunan

Rentang Nilai (n)	Tingkat Kerusakan
$n < 0$	Rusak Ringan
$0 \leq n < 2$	Rusak Sedang
$n \geq 2$	Rusak Berat

Sumber : (Almais et al., 2024)

Jika nilai PC1 kurang dari 0, data dikategorikan sebagai "Rusak Ringan". Jika nilai PC1 berada di rentang 0 hingga kurang dari 2, data dikategorikan sebagai

"Rusak Sedang". Jika nilai PC lebih besar atau sama dengan 2, data dikategorikan sebagai "Rusak Berat". Kategori hasil klasifikasi kemudian ditambahkan ke dalam DataFrame `principalDF` pada kolom baru bernama `Kategori`. Setiap baris data akan memiliki label kategori yang sesuai dengan nilai PC-nya.

Tabel 4. 6 Hasil PC1 yang sudah dinormalisasi

Index	PC1	Kategori
0	4.000000	Rusak Berat
1	2.518215	Rusak Berat
2	2.868550	Rusak Berat
3	3.088221	Rusak Berat
4	3.338613	Rusak Berat
5	2.275165	Rusak Berat
6	-1.159660	Rusak Ringan
7	-1.279610	Rusak Ringan
8	-1.482524	Rusak Ringan
9	1.646636	Rusak Sedang
10	2.361320	Rusak Berat
11	-1.466478	Rusak Ringan
12	-1.342997	Rusak Ringan
13	-1.554859	Rusak Ringan
14	1.245206	Rusak Sedang
15	2.532612	Rusak Berat
16	-1.335282	Rusak Ringan
17	-1.813871	Rusak Ringan
18	-2.000000	Rusak Ringan
19	0.333084	Rusak Sedang
20	3.344756	Rusak Berat
21	1.057395	Rusak Sedang
22	1.602003	Rusak Sedang
23	1.000316	Rusak Sedang
24	2.636063	Rusak Berat

Tabel 4.6 menunjukkan hasil nilai PC1 yang telah dinormalisasi dalam rentang -2 hingga 4, yang kemudian diklasifikasikan menjadi tiga kategori: Rusak Ringan, Rusak Sedang, dan Rusak Berat. Kategori ini ditentukan berdasarkan batas nilai PC1 yang telah diatur, yaitu nilai di bawah 0 untuk Rusak Ringan, 0 hingga

kurang dari 2 untuk Rusak Sedang, dan nilai 2 ke atas untuk Rusak Berat. Proses normalisasi ini bertujuan untuk menyelaraskan distribusi data agar lebih mudah diinterpretasikan dalam analisis tingkat kerusakan.

Tabel 4. 7 Titik Hasil Rusak

Hasil rusak berdasarkan jumlah titik terbanyak	
Jumlah titik untuk Rusak Ringan	9
Jumlah titik untuk Rusak Sedang	6
Jumlah titik untuk Rusak Berat	10

Tabel 4.7 menunjukkan hasil klasifikasi tingkat kerusakan bangunan berdasarkan jumlah titik terbanyak yang berada dalam kategori tertentu. Berdasarkan hasil analisis, jumlah titik untuk kategori *Rusak Berat* sebanyak 10 titik, menjadikannya kategori dengan jumlah titik terbanyak dibandingkan dengan kategori lainnya. Kategori *Rusak Ringan* memiliki 9 titik, sementara *Rusak Sedang* memiliki jumlah titik paling sedikit, yaitu 6 titik. Hasil ini mencerminkan distribusi tingkat kerusakan yang didominasi oleh kategori *Rusak Berat*, yang dapat diartikan bahwa mayoritas bangunan dalam dataset mengalami tingkat kerusakan yang signifikan. Analisis ini mendukung pengambilan keputusan dalam prioritas penanganan kerusakan pascabencana.

E. Hasil Visualisasi

Visualisasi data merupakan langkah penting dalam menyampaikan hasil analisis data secara intuitif. Dalam konteks penelitian ini, visualisasi digunakan untuk memetakan kategori kerusakan bangunan pasca bencana berdasarkan hasil analisis *Principal Component Analysis* (PCA). Dengan memanfaatkan komponen utama PCA (PC1, PC2, dan PC3), data yang telah dikategorikan menjadi "Rusak

Ringan," "Rusak Sedang," dan "Rusak Berat" divisualisasikan untuk memberikan pemahaman yang lebih jelas mengenai distribusi dan pola kerusakan.

Proses pengambilan keputusan dilakukan dengan menggunakan nilai PC1 yang telah dinormalisasi ke rentang -2 hingga 4. Berdasarkan nilai PC1, kategori kerusakan ditentukan: nilai PC1 kurang dari 0 diklasifikasikan sebagai "Rusak Ringan," nilai antara 0 hingga kurang dari 2 sebagai "Rusak Sedang," dan nilai lebih dari atau sama dengan 2 sebagai "Rusak Berat." Kolom kategori ini kemudian ditambahkan ke dalam DataFrame utama, bersama nilai komponen PC1, PC2, dan PC3, yang akan digunakan dalam visualisasi.

Pseudocode memanfaatkan nilai PC1, PC2, dan PC3 untuk menghasilkan visualisasi yang relevan. Dalam visualisasi 2D, scatter plot dibuat menggunakan koordinat PC1 dan PC2, sementara kategori kerusakan dimapping ke warna tertentu melalui fungsi map. Untuk visualisasi 3D, nilai PC1, PC2, dan PC3 digunakan untuk memplot data dalam ruang tiga dimensi dengan modul `mpl_toolkits.mplot3d`. Titik-titik data diberi warna berdasarkan kategorinya, yang mencerminkan hasil pengelompokan. Proses ini menghasilkan visualisasi yang mudah dipahami untuk menganalisis distribusi kategori kerusakan berdasarkan dimensi PCA.

```
Function Visualize2DPCA(principalDF, colors, hasil_rusak):
    Input:  principalDF    (DataFrame),    colors    (Dictionary),
    hasil_rusak (String)

    Initialize Figure with dimensions (10, 6)

    For each kategori, color in colors:
        Subset data where 'Kategori' equals kategori
```

```

        Scatter plot for subset with PC1 values on x-axis, y-
values set to 0, use specified color

        Set x-axis label to 'PC1'

        Remove y-axis labels

        Set plot title to 'PCA - Scatter Plot Berdasarkan PC1 dan
Kategori Rusak (Hasil: hasil_rusak)'

        Set x-axis range to (-2, 4)

        Enable grid with 'x' axis only

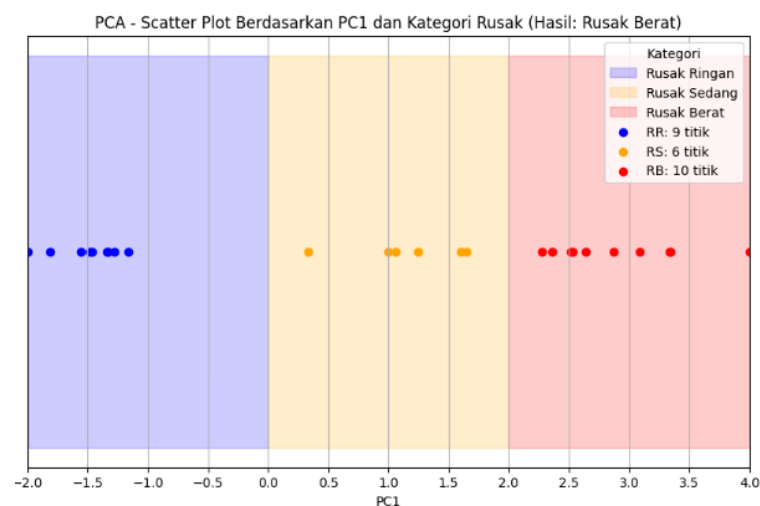
        Add legend with title 'Kategori'

        Show the plot

End Function

```

Hasil visualisasi yang ditampilkan dalam dua grafik ini memberikan representasi visual dari analisis komponen utama (PCA) dan kategori kerusakan yang dihasilkan berdasarkan nilai PC1.



Gambar 4. 7 Visualisasi Hasil Kategori

Visualisasi ini menampilkan hubungan antara dua komponen utama pertama, yaitu PC1. Setiap titik pada grafik mewakili satu data yang telah diproses menggunakan PCA. Warna titik-titik tersebut menggambarkan kategori tingkat kerusakan bangunan berdasarkan nilai PC1, yang telah di-normalisasi. Warna biru menunjukkan kategori "Rusak Ringan", yang berada pada rentang $n < 0$. Warna oranye menunjukkan kategori "Rusak Sedang", berada di antara $0 \leq n < 2$. Warna merah menunjukkan kategori "Rusak Berat", dengan $n \geq 2$. Pola distribusi ini menunjukkan bagaimana data terkelompok berdasarkan tingkat kerusakan, dengan data pada kategori "Rusak Berat" terkonsentrasi di kanan atas grafik (nilai PC1 tinggi), sedangkan kategori "Rusak Ringan" berada di bagian kiri (nilai PC1 rendah).

4.2.3 Augmentasi Data

1.) Reflected

Augmentasi data dilakukan dengan tujuan meningkatkan variasi dataset tanpa menambah jumlah data asli. Pada penelitian ini, salah satu metode augmentasi yang digunakan adalah refleksi horizontal (flip horizontal), di mana citra asli diubah dengan membalik posisi pikselnya pada sumbu vertikal (kiri menjadi kanan dan

sebaliknya). Teknik ini dapat membantu model pembelajaran mesin dalam memahami fitur yang lebih general dari berbagai orientasi gambar.



Gambar 4. 8 Proses Augmentasi Reflection

Hasil statistik dari citra refleksi dapat dilihat pada Tabel dibawah ini, yang menunjukkan nilai *Mean*, *Variance*, *Entropy*, *Energy*, dan *Skewness* untuk setiap grid dari citra hasil refleksi. Berdasarkan hasil analisis, distribusi nilai-nilai ini tidak menunjukkan perubahan yang signifikan dibandingkan data asli, namun tetap memberikan variasi yang relevan untuk keperluan pelatihan model.

Tabel 4. 8 Hasil Ekstraksi Reflection

Index	Mean	Variance	Entropy	Energy	Skewness
0	10.265537	440.751524	2.087596	0.498999	7.648406
1	13.396396	537.053681	2.525353	0.375174	7.473458
2	14.112132	563.301765	2.535366	0.367478	7.411698
3	14.624074	578.001272	2.61177	0.355734	7.38664
4	12.946629	483.283668	2.695645	0.362348	7.479495
5	14.836364	552.675041	2.833479	0.326869	7.366824
6	24.032698	662.27141	3.907622	0.109593	4.659848
7	22.447028	625.301458	4.031876	0.106197	4.888503
8	24.416901	704.395207	3.579135	0.150946	5.651663
9	19.709278	665.92579	3.073567	0.2486	6.979498
10	14.464158	551.184199	2.735499	0.343771	7.394258
11	25.857542	669.312108	4.017044	0.098343	4.423449
12	26.347953	724.683022	3.706183	0.120242	4.696329
13	23.507653	657.831574	4.022276	0.105932	4.767877
14	18.020576	624.797931	3.097749	0.256871	7.036764
15	19.14902	691.864068	2.645397	0.335825	7.246422
16	29.773006	729.181603	3.654479	0.112161	3.921052

Index	Mean	Variance	Entropy	Energy	Skewness
17	28.188623	729.123104	3.776641	0.11345	4.105596
18	25.03183	704.513576	3.798298	0.126104	5.206398
19	16.498998	596.478456	3.044223	0.266911	7.103475
20	11.916185	486.90916	2.484557	0.409661	7.556128
21	19.176829	667.625235	3.037403	0.266574	7.075316
22	16.619417	584.519235	3.176022	0.235954	6.916325
23	18.397089	661.794503	2.921747	0.27847	7.082879
24	13.391376	512.785465	2.638872	0.350627	7.373708

Komponen utama pertama (PC1) berhasil menjelaskan 90.96% dari total varians data. Hal ini menunjukkan bahwa sebagian besar informasi penting dari dataset dapat direpresentasikan oleh dimensi pertama. Eigen Value untuk komponen pertama (PC1) adalah 4.74, yang menunjukkan seberapa besar kontribusi PC1 dalam mendeskripsikan varians total data. Semakin besar nilai Eigen Value, semakin signifikan kontribusi komponen terhadap varians data.

Nilai *Explained Variance Ratio* yang tinggi pada PC1 mengindikasikan bahwa variabel-variabel asli memiliki korelasi tinggi, sehingga informasi mereka dapat diringkas ke dalam satu dimensi utama tanpa kehilangan banyak informasi. Nilai Eigen Value lebih besar dari 1 (4.74) pada PC1 menguatkan bahwa dimensi pertama adalah dimensi yang paling relevan untuk merepresentasikan dataset.

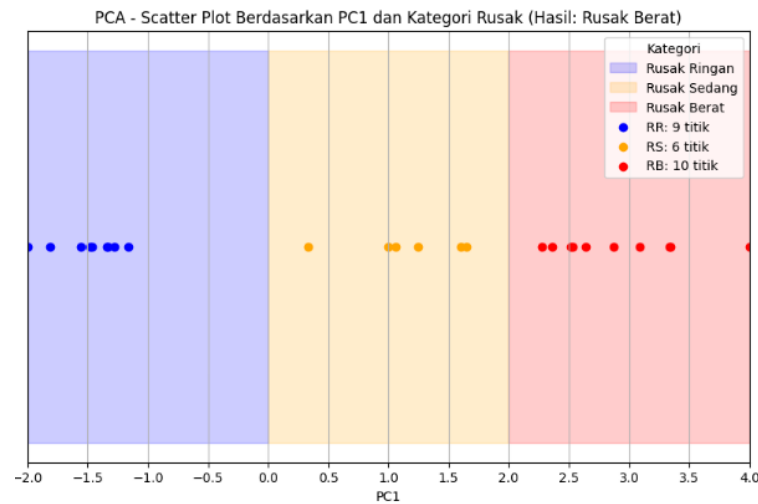
Tabel 4. 9 Hasil Titik Reflection

Hasil rusak berdasarkan jumlah titik terbanyak	
Jumlah titik untuk Rusak Ringan	9
Jumlah titik untuk Rusak Sedang	6
Jumlah titik untuk Rusak Berat	10

Berdasarkan hasil clustering dan normalisasi nilai PC1, data dibagi menjadi tiga kategori kerusakan:

- Rusak Ringan: Sebanyak 9 titik (contoh: *Mean* rendah dan *Entropy* rendah).

- Rusak Sedang: Sebanyak 6 titik (contoh: *Variance* sedang dan *Energy* sedang).
- Rusak Berat: Sebanyak 10 titik (contoh: *Skewness* rendah dan *Variance* tinggi).



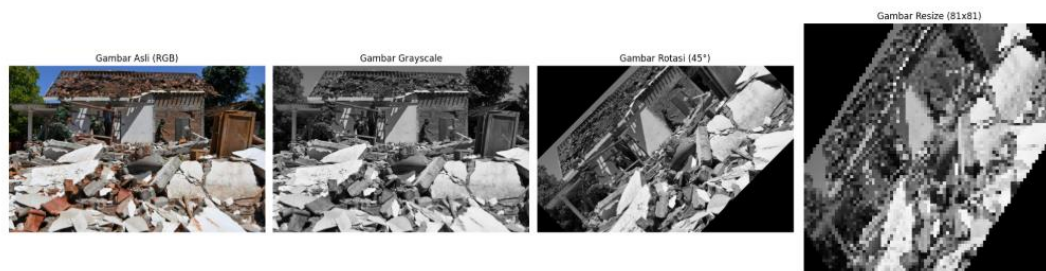
Gambar 4. 9 Hasil Visualisasi Reflection

Proses augmentasi refleksi memberikan kontribusi yang signifikan terhadap peningkatan variasi data tanpa mengubah struktur statistik dasar citra. Teknik ini efektif dalam memperkaya dataset untuk melatih model pembelajaran mesin agar lebih robust terhadap variasi orientasi gambar. Dengan hasil ini, model diharapkan dapat mengenali pola kerusakan bangunan dengan lebih baik, meskipun citra memiliki orientasi yang berbeda.

Lalu, proses augmentasi pada citra dilakukan untuk meningkatkan variasi data dengan tujuan menguji konsistensi model analisis terhadap perubahan orientasi gambar.

2.) Rotated

Rotasi citra merupakan salah satu teknik praproses dalam pengolahan gambar yang digunakan untuk mengubah orientasi gambar, seperti memutar sebesar 45 derajat, dengan tujuan memastikan fitur tekstur tetap dapat dianalisis secara optimal menggunakan metode PCA (Principal Component Analysis) dalam menentukan tingkat kerusakan bangunan.



Gambar 4. 10 Rotated

Dalam kasus ini, citra telah dirotasi sebesar 45 derajat, kemudian dilakukan analisis menggunakan metode PCA (*Principal Component Analysis*) untuk mereduksi dimensi dan mengevaluasi tingkat kerusakan bangunan berdasarkan fitur tekstur.

Tabel 4. 10 Hasil Ekstraksi Rotated

Index	Mean	Variance	Entropy	Energy	Skewness
0	21.807692	898.309172	0.930586	0.547337	6.104197
1	20.898635	830.161265	1.947211	0.389297	6.017433
2	13.735780	557.060463	2.450350	0.387648	7.447109
3	15.144954	585.536786	2.637208	0.355354	7.345877
4	16.479560	664.306186	2.218481	0.417794	7.277665
5	27.775510	961.957768	1.476495	0.439334	5.472727

Index	Mean	Variance	Entropy	Energy	Skewness
6	25.278788	732.413186	3.568283	0.136253	4.943525
7	26.871429	663.649184	4.030530	0.093910	3.977449
8	26.747191	699.851818	3.873244	0.107452	4.263316
9	21.415755	732.824960	2.918807	0.263899	6.723826
10	15.461676	620.355483	2.599419	0.333232	6.939653
11	26.959420	723.859223	3.732405	0.122705	4.463342
12	28.510972	731.723234	3.639872	0.118149	4.117902
13	29.512121	749.310459	3.709525	0.117062	4.155426
14	21.921687	753.469771	3.003750	0.248698	6.442240
15	15.304425	585.602901	2.798140	0.335146	7.341496
16	24.886792	706.440014	3.888045	0.116397	4.793622
17	25.120787	665.578107	3.932425	0.104280	4.473159
18	23.605405	633.130782	4.122579	0.093543	4.262154
19	31.593548	931.987485	2.034393	0.364690	5.348832
20	14.495495	601.631361	2.262455	0.429290	7.387248
21	18.285429	645.760846	3.075114	0.258354	7.036809
22	17.975104	630.119712	3.016374	0.264863	6.983353
23	25.443737	857.414563	2.509049	0.300192	5.728966
24	29.538462	988.402367	0.997201	0.501939	5.422429

Setelah rotasi 45 derajat, dilakukan pengukuran statistik deskriptif berupa *Mean*, *Variance*, *Entropy*, *Energy*, dan *Skewness*. Hasil menunjukkan bahwa fitur-fitur tekstur dari citra yang dirotasi memberikan nilai yang konsisten dengan data awal, seperti:

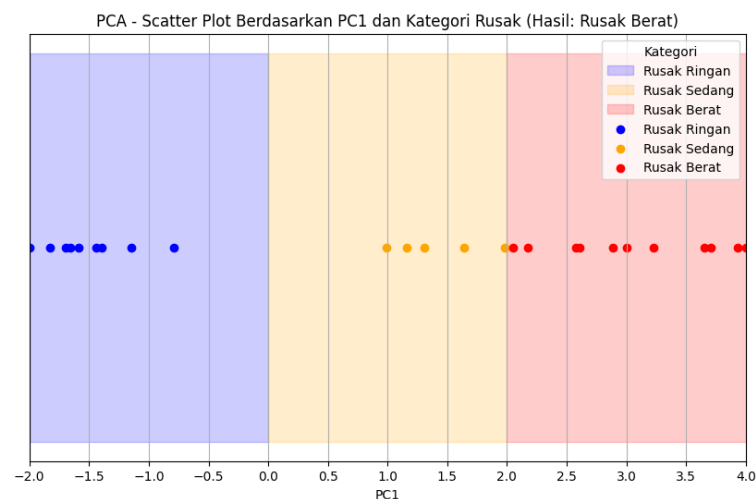
- *Mean* menunjukkan nilai intensitas rata-rata.
- *Variance* menyoroti tingkat sebaran intensitas, dengan nilai tertinggi sebesar 988.40.
- *Entropy* mengukur tingkat ketidakaturan, dengan nilai tertinggi 4.12.
- *Energy* menunjukkan densitas informasi pada citra.
- *Skewness* menggambarkan simetri distribusi intensitas, dengan nilai tertinggi 7.44.

Komponen utama pertama menjelaskan 56.46% variabilitas data. Artinya, sebagian besar informasi dalam data dapat dirangkum ke dalam komponen pertama, sehingga efektif untuk reduksi dimensi. Nilai *eigen* sebesar 2.94, menunjukkan pentingnya komponen utama dalam menggambarkan pola data.

Tabel 4. 11 Hasil Titik Rotated

Hasil rusak berdasarkan jumlah titik terbanyak	
Jumlah titik untuk Rusak Ringan	9
Jumlah titik untuk Rusak Sedang	5
Jumlah titik untuk Rusak Berat	11

Hasil analisis menunjukkan bahwa meskipun citra dirotasi sebesar 45 derajat, nilai-nilai statistik dan hasil klasifikasi kerusakan tetap konsisten dengan data awal. Hal ini mengindikasikan bahwa metode yang digunakan mampu mempertahankan informasi penting dalam citra, sehingga rotasi tidak memengaruhi hasil deteksi kerusakan.

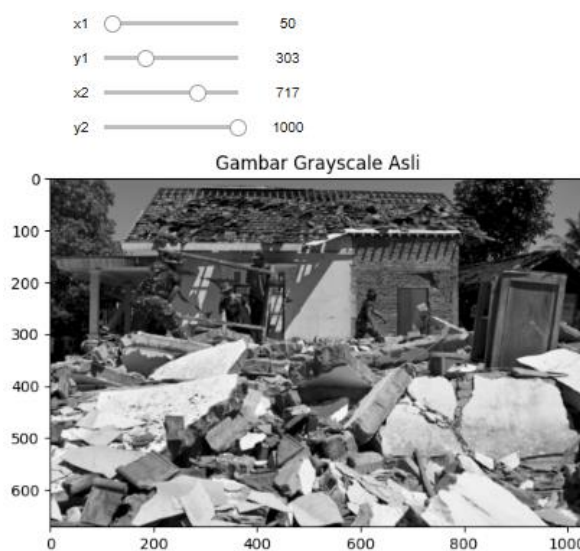


Gambar 4. 11 Hasil Visualisasi Rotated

Rotasi 45 derajat pada citra tidak mengubah hasil analisis kerusakan. Dengan demikian, metode ini menunjukkan kestabilan terhadap perubahan orientasi

gambar, yang penting dalam aplikasi dunia nyata untuk mendeteksi kerusakan bangunan dengan sudut pandang berbeda.

3.) *Scalled*



Gambar 4. 12 Grayscale Scalling

Gambar di atas merupakan hasil penelitian yang menunjukkan proses seleksi area fokus pada citra grayscale bangunan pasca bencana menggunakan koordinat interaktif yang telah ditentukan. Dengan parameter $x1=50$, $y1=303$, $x2=717$, dan $y2=1000$, area yang dipilih berfokus pada bagian tengah hingga bawah bangunan yang mengalami kerusakan signifikan. Pemilihan area ini bertujuan untuk menyoroti bagian kerusakan yang dominan, sehingga dapat diolah lebih lanjut menggunakan metode Local Binary Pattern (LBP) untuk ekstraksi fitur tekstur dan Principal Component Analysis (PCA) untuk analisis klasifikasi tingkat kerusakan



Gambar 4. 13 Hasil Zoom

Gambar di atas merupakan hasil zoom dari area tertentu pada citra grayscale bangunan yang telah dipilih berdasarkan koordinat spesifik. Fokus pada bagian ini menampilkan detail kerusakan struktur bangunan, seperti retakan, reruntuhan material, dan pecahan dinding yang lebih jelas.

Tabel 4. 12 Ekstraksi Fitu Scalled

Index	Mean	Variance	Entropy	Energy	Skewness
0	11.641667	491.721597	2.093029	0.500937	7.674898
1	16.895795	647.585119	2.703083	0.329759	7.249515
2	15.280289	587.731564	2.759998	0.337665	7.378112
3	17.638941	673.761886	2.566077	0.348537	7.289579
4	13.050746	534.591455	2.397668	0.425663	7.586514
5	15.181982	579.727243	2.724425	0.331295	7.305237
6	25.870588	695.218547	4.019244	0.095156	4.204806
7	26.356955	693.358147	3.925207	0.110960	4.686112
8	30.216374	690.567217	4.092285	0.084590	3.116357
9	18.786008	660.003590	2.922042	0.270614	6.997656
10	15.714286	598.304267	2.773420	0.335139	7.384929
11	30.495114	756.771149	3.537123	0.123959	3.924598
12	26.884181	691.339693	3.907387	0.108063	4.570683
13	26.578035	730.093622	3.692410	0.128170	5.057343
14	20.841871	702.694372	2.918875	0.260133	6.820703
15	14.612431	538.427487	2.918629	0.306344	7.332198
16	23.615190	622.935465	4.188015	0.085365	3.545276
17	23.770308	690.826793	3.642088	0.135952	5.238103
18	25.775561	619.151622	4.222148	0.086511	4.164823
19	18.080082	592.878597	3.238983	0.237514	7.015236
20	13.242515	520.884300	2.573413	0.391956	7.528957
21	18.719124	632.628280	3.207666	0.231346	6.860189

Index	Mean	Variance	Entropy	Energy	Skewness
22	19.230453	676.613558	3.079385	0.242316	6.789069
23	18.041152	614.763739	2.981516	0.268286	7.084404
24	14.900482	545.184318	2.842351	0.332382	7.442326

Tabel di atas menyajikan hasil ekstraksi fitur Local Binary Pattern (LBP) dari citra bangunan yang diolah menggunakan metode pengolahan citra. Setiap baris pada tabel merepresentasikan indeks area gambar (Index) yang telah diolah, sementara kolom-kolom menyajikan fitur statistik yang dihasilkan, yaitu:

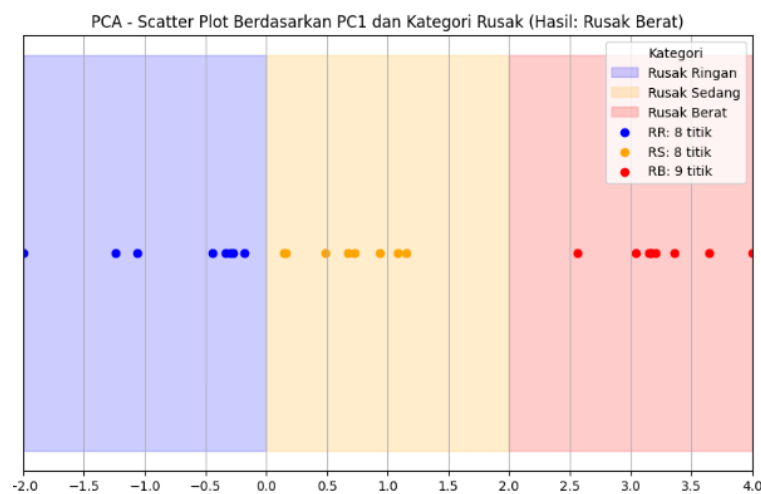
1. Mean: Rata-rata nilai LBP yang mencerminkan tingkat intensitas rata-rata pada area gambar. Nilai ini berkisar antara 11.64 hingga 30.49, menunjukkan variasi distribusi kecerahan pada area yang dipilih.
2. Variance: Menunjukkan tingkat penyebaran nilai intensitas piksel. Variance yang tinggi, seperti 756.77, menunjukkan bahwa area memiliki variasi kecerahan yang signifikan, sedangkan nilai rendah menunjukkan distribusi yang lebih seragam.
3. Entropy: Mengukur tingkat ketidakberaturan atau kompleksitas tekstur dalam area gambar. Nilai entropy tertinggi berada di 4.22, yang menandakan area tersebut memiliki tekstur yang lebih acak atau kompleks.
4. Energy: Mengukur kehalusan tekstur, di mana nilai energy rendah (misalnya 0.084) menunjukkan tekstur yang lebih kasar atau kompleks, sedangkan energy tinggi menunjukkan tekstur lebih seragam dan halus.
5. Skewness: Mengukur asimetri distribusi nilai piksel. Nilai skewness yang tinggi, seperti 7.67, menunjukkan bahwa distribusi intensitas piksel pada area

cenderung miring ke satu arah, mencerminkan adanya dominasi piksel terang atau gelap.

Tabel 4. 13 Jumlah Titik Scalled

Hasil rusak berdasarkan jumlah titik terbanyak	
Jumlah titik untuk Rusak Ringan	8
Jumlah titik untuk Rusak Sedang	8
Jumlah titik untuk Rusak Berat	9

Berdasarkan tabel dan scatter plot di atas, hasil klasifikasi tingkat kerusakan bangunan dilakukan menggunakan metode Principal Component Analysis (PCA). Tabel menunjukkan jumlah titik untuk setiap kategori kerusakan, yakni 8 titik untuk Rusak Ringan, 8 titik untuk Rusak Sedang, dan 9 titik untuk Rusak Berat. Jumlah titik ini menentukan distribusi data pada masing-masing kategori.



Gambar 4. 14 Visualisasi Scalling

Pada plot tersebut, area berwarna biru mewakili kategori Rusak Ringan, area oranye untuk Rusak Sedang, dan area merah untuk Rusak Berat. Titik-titik berwarna biru, oranye, dan merah menunjukkan penyebaran data sesuai kategori yang telah diklasifikasikan. Rusak Berat memiliki jumlah titik terbanyak, yakni 9

titik, yang tersebar dominan pada area merah (nilai $PC1 \geq 2$). Sementara itu, Rusak Ringan dan Rusak Sedang masing-masing memiliki 8 titik yang tersebar merata di area biru ($PC1 < 0$) dan oranye ($0 \leq PC1 < 2$).

Pada proses scaling menggunakan metode PCA (Principal Component Analysis), terjadi perubahan distribusi titik pada scatter plot akibat normalisasi nilai $PC1$. Perubahan ini terjadi karena data citra kehilangan sebagian informasi akibat reduksi dimensi, yang mengakibatkan pergeseran posisi titik. Meskipun jumlah titik untuk masing-masing kategori Rusak Ringan, Rusak Sedang, dan Rusak Berat mengalami sedikit pergeseran dalam visualisasi, hasil klasifikasi tetap konsisten. Hal ini menunjukkan bahwa sistem masih mampu mempertahankan pola distribusi data yang mencerminkan tingkat kerusakan sesuai kategori awal. Kehilangan informasi dalam proses reduksi dimensi PCA menjadi konsekuensi yang umum terjadi, namun klasifikasi tetap akurat karena pola utama dalam data masih dapat dipertahankan dengan baik.

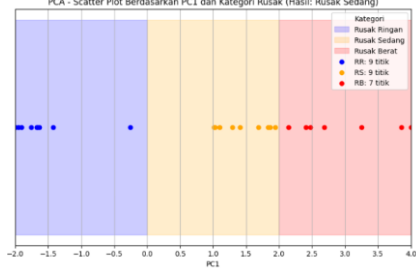
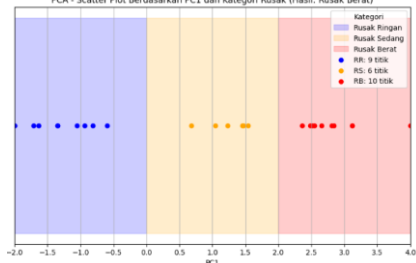
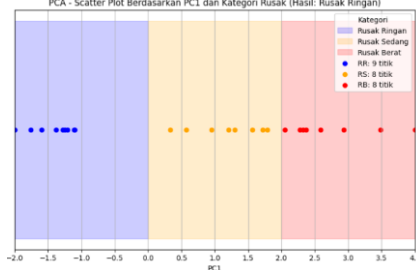
4.3 Pembahasan

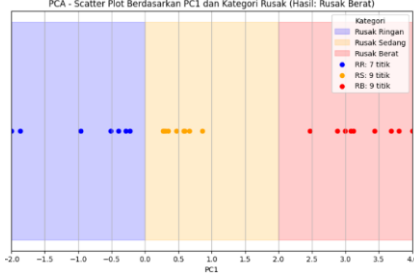
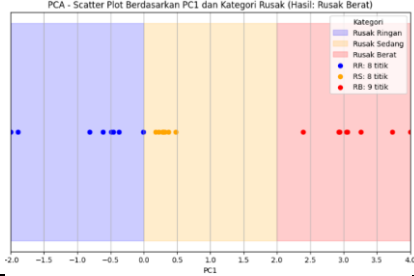
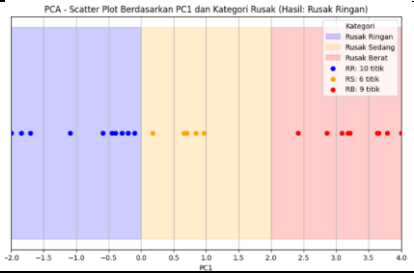
Uji coba sistem ini menggunakan 12 gambar kerusakan bangunan dari data BPBD Kota Malang untuk mengidentifikasi tingkat kerusakan bangunan melalui penerapan metode *Local Binary Pattern* (LBP) dan *Principal Component Analysis* (PCA). Proses dimulai dengan pra-pemrosesan gambar, di mana gambar RGB dikonversi menjadi *grayscale* dan di-*resize* ke ukuran 81x81 piksel untuk mempermudah analisis. Pada tahap *thresholding*, pola biner dihasilkan dari perbandingan nilai intensitas piksel pusat dengan tetangganya, yang selanjutnya

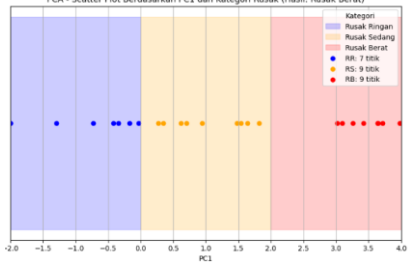
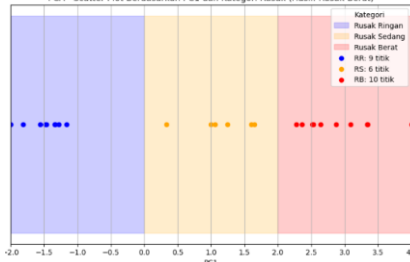
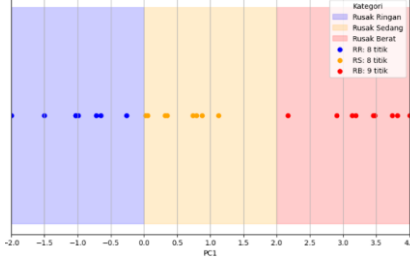
dikonversi menjadi nilai numerik LBP untuk menggambarkan tekstur lokal pada radius 1 hingga 5. Setelah itu, data dinormalisasi menggunakan StandardScaler untuk menyamakan skala setiap fitur seperti *Mean*, *Variance*, *Entropy*, *Energy*, dan *Skewness*. PCA kemudian diterapkan untuk mereduksi dimensi data menjadi satu komponen utama, yaitu PC1.

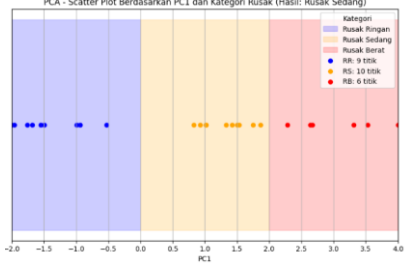
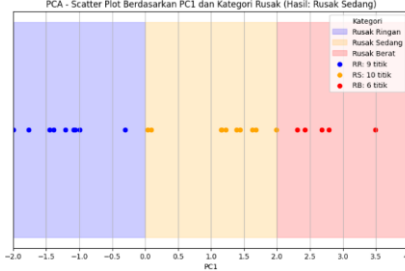
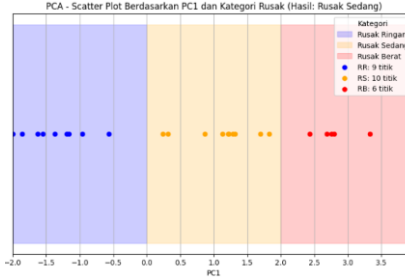
Pada percobaan yang sama, sistem juga diuji dengan menggunakan gambar yang telah di-augmentasi melalui proses flip horizontal, dan rotasi sebesar 45 derajat untuk menguji kestabilan model terhadap perubahan orientasi dan posisi gambar. Hasil analisis menunjukkan bahwa pola tekstur yang dihasilkan tetap konsisten, baik pada data asli maupun data yang telah diubah melalui augmentasi. Proses ekstraksi fitur menggunakan metode LBP dan reduksi dimensi dengan PCA berhasil mempertahankan informasi utama dari gambar, sehingga hasil identifikasi tingkat kerusakan tidak terpengaruh oleh perubahan orientasi dan posisi gambar. Hal ini menunjukkan bahwa sistem yang dikembangkan memiliki robustnes yang tinggi, sehingga dapat diandalkan untuk analisis gambar dengan variasi transformasi geometris.

Tabel 4. 14 Hasil Pengujian

No	Gambar asli	Grayscale	Threshold	Hasil Label	Hasil Uji Sistem	Penilaian Pakar
1.				PCA - Scatter Plot Berdasarkan PC1 dan Kategori Rusak (Hasil: Rusak Sedang) 	Rusak Sedang	Rusak Sedang
2.				PCA - Scatter Plot Berdasarkan PC1 dan Kategori Rusak (Hasil: Rusak Berat) 	Rusak Berat	Rusak Berat
3.				PCA - Scatter Plot Berdasarkan PC1 dan Kategori Rusak (Hasil: Rusak Ringan) 	Rusak Ringan	Rusak Ringan

No	Gambar asli	Grayscale	Threshold	Hasil Label	Hasil Uji Sistem	Penilaian Pakar
4.				PCA - Scatter Plot Berdasarkan PC1 dan Kategori Rusak (Hasil: Rusak Berat) 	Rusak Berat	Rusak Berat
5.				PCA - Scatter Plot Berdasarkan PC1 dan Kategori Rusak (Hasil: Rusak Berat) 	Rusak Berat	Rusak Berat
6.				PCA - Scatter Plot Berdasarkan PC1 dan Kategori Rusak (Hasil: Rusak Ringan) 	Rusak Ringan	Rusak Ringan

No	Gambar asli	Grayscale	Threshold	Hasil Label	Hasil Uji Sistem	Penilaian Pakar
7.				PCA - Scatter Plot Berdasarkan PC1 dan Kategori Rusak (Hasil: Rusak Berat) 	Rusak Sedang	Rusak Sedang
8.				PCA - Scatter Plot Berdasarkan PC1 dan Kategori Rusak (Hasil: Rusak Berat) 	Rusak Berat	Rusak Berat
9.				PCA - Scatter Plot Berdasarkan PC1 dan Kategori Rusak (Hasil: Rusak Berat) 	Rusak Berat	Rusak Berat

No	Gambar asli	Grayscale	Threshold	Hasil Label	Hasil Uji Sistem	Penilaian Pakar
10.				PCA - Scatter Plot Berdasarkan PC1 dan Kategori Rusak (Hasil: Rusak Sedang) 	Rusak Sedang	Rusak Sedang
11.				PCA - Scatter Plot Berdasarkan PC1 dan Kategori Rusak (Hasil: Rusak Sedang) 	Rusak Sedang	Rusak Sedang
12.				PCA - Scatter Plot Berdasarkan PC1 dan Kategori Rusak (Hasil: Rusak Sedang) 	Rusak Sedang	Rusak Sedang

Berdasarkan tabel (4.14) merupakan hasil analisis 12 data citra bangunan pasca bencana alam yang diklasifikasikan berdasarkan tingkat kerusakan menggunakan metode PCA (*Principal Component Analysis*). Setiap citra melalui beberapa tahapan pengolahan, meliputi Gambar Asli dalam format RGB, Grayscale untuk menyederhanakan informasi warna, dan *Thresholding* sebagai proses binerisasi untuk mendeteksi area yang menunjukkan kerusakan struktur bangunan. Visualisasi hasil menggunakan scatter plot PCA berdasarkan nilai PC1 yang dinormalisasi menunjukkan distribusi data yang dikategorikan ke dalam Rusak Ringan (biru), Rusak Sedang (oranye), dan Rusak Berat (merah).

Berdasarkan hasil uji sistem dan validasi penilaian pakar, klasifikasi tingkat kerusakan menunjukkan konsistensi. Gambar 2, 4, 5, 8, dan 9 dikategorikan sebagai Rusak Berat dengan dominasi distribusi titik pada area merah, yang menunjukkan keruntuhan total atau signifikan. Gambar 3, 6 dikategorikan sebagai Rusak Ringan, dengan distribusi titik pada area biru yang menggambarkan kerusakan minor atau sedikit pergeseran struktur. Sementara itu, gambar 1, 7, 10, 11, dan 12 dikategorikan sebagai Rusak Sedang, yang ditunjukkan oleh distribusi titik di area oranye yang menandakan kerusakan parsial tetapi struktur masih berdiri tegak. Analisis ini menegaskan bahwa sistem memiliki tingkat akurasi tinggi karena hasil klasifikasi sejalan dengan penilaian pakar.

Gambar 7, Gambar 4, dan Gambar 1 menunjukkan distribusi titik yang dihasilkan dari analisis PCA untuk menentukan tingkat kerusakan bangunan. Apabila jumlah titik pada kategori Rusak Ringan (RR), Rusak Sedang (RS), dan Rusak Berat (RB) sama, maka label yang dipilih adalah kategori di mana sebaran

titik tersebut paling merata dan berada di area yang konsisten dengan kategori tersebut. Hal ini memastikan bahwa penentuan label kerusakan tidak hanya berdasarkan jumlah titik, tetapi juga memperhitungkan sebaran data pada scatter plot. Dalam contoh ini, analisis visual menunjukkan bahwa titik yang tersebar merata di area tertentu memberikan indikasi yang lebih kuat untuk menentukan tingkat kerusakan dibandingkan dengan titik yang terkonsentrasi di satu area saja.

Dalam pengolahan gambar, dilakukan penilaian skor validasi clustering untuk mengevaluasi kualitas hasil pengelompokan tingkat kerusakan. Tabel 4.15 menyajikan Skor Silhouette Score yang menunjukkan sejauh mana data dalam setiap kategori (Rusak Ringan, Rusak Sedang, dan Rusak Berat) terbentuk dengan baik berdasarkan metode PCA yang digunakan.

Tabel 4. 15 Skor Validasi

No	Keterangan	Label	Skor Silhoutte Score
1	Gambar 1	Rusak Sedang	0.6241337412020467
2	Gambar 2	Rusak Berat	0.7318451045515799
3	Gambar 3	Rusak Ringan	0.6363767698950812
4	Gambar 4	Rusak Berat	0.659718426622899
5	Gambar 5	Rusak Berat	0.645471426146336
6	Gambar 6	Rusak Ringan	0.6146788557125129
7	Gambar 7	Rusak Sedang	0.6261511672796363
8	Gambar 8	Rusak Berat	0.7222229852302833
9	Gambar 9	Rusak Berat	0.6589783059692274
10	Gambar 10	Rusak Sedang	0.6924081183235473
11	Gambar 11	Rusak Sedang	0.586163860657802
12	Gambar 12	Rusak Sedang	0.6842269643914234

Tabel 4.14 menyajikan hasil validasi *clustering* pada 12 gambar yang diklasifikasikan berdasarkan tingkat kerusakan dengan menggunakan Skor *Silhouette Score*. Skor ini digunakan untuk mengevaluasi seberapa baik data dalam setiap cluster terbentuk dan terpisah satu sama lain. Hasil pada tabel menunjukkan

bahwa gambar dengan label Rusak Berat memiliki skor tertinggi sebesar 0.7318 (Gambar 2) yang mengindikasikan kualitas clustering yang baik untuk kategori tersebut. Sementara itu, gambar dengan label Rusak Sedang memiliki skor berkisar antara 0.5861 hingga 0.6924, yang menunjukkan bahwa kategori ini memiliki tingkat kejelasan yang cukup baik namun lebih bervariasi. Untuk kategori Rusak Ringan, skor berkisar pada 0.6146 hingga 0.6363, yang menandakan pengelompokan kategori ini cukup stabil. Secara keseluruhan, skor validasi ini menunjukkan bahwa metode clustering mampu mengkategorikan gambar ke dalam tiga kategori kerusakan: kerusakan ringan, kerusakan sedang, dan kerusakan berat.

4.4 Integrasi Islam

4.4.1 *Muamalah Ma'a Allah*

Hubungan manusia dengan Allah, atau *muamalah ma'a Allah*, adalah pilar utama dalam ajaran Islam untuk menjalani kehidupan, termasuk menghadapi bencana. Musibah seperti bencana alam adalah bagian dari takdir Allah yang harus diterima dengan ikhlas dan kesabaran. Dalam QS. Al-Baqarah ayat 153 disebutkan:

يَا أَيُّهَا الَّذِينَ آمَنُوا اسْتَعِينُوا بِالصَّبْرِ وَالصَّلَاةِ إِنَّ اللَّهَ مَعَ الصَّابِرِينَ

"Hai orang-orang yang beriman, jadikanlah sabar dan sholat sebagai penolongmu, sesungguhnya Allah beserta orang-orang yang sabar." (Q.S. Al-Baqarah: 153).

Dalam tafsir Al-Muyassar, dijelaskan bahwa dua cara utama untuk meminta pertolongan Allah dalam semua hal, termasuk saat menghadapi musibah, adalah kesabaran dan sholat. Oleh karena itu, manusia diperintahkan untuk tetap menjalani ujian yang diberikan dengan penuh keikhlasan dan memperkuat ibadahnya. Dalam

konteks penelitian ini, teknologi yang dikembangkan diharapkan menjadi salah satu bentuk ikhtiar manusia dalam menghadapi kerusakan akibat bencana, disertai dengan kesadaran bahwa segala sesuatu terjadi atas kehendak Allah. Manusia hanya bisa berusaha dan berdoa untuk mendapatkan pertolongan-Nya.

4.4.2 *Muamalah Ma'a An-Nas*

Islam juga sangat menekankan pentingnya hubungan antar manusia atau *muamalah ma'a an-nas*, terutama dalam bentuk tolong-menolong. Penelitian ini bertujuan untuk membantu sesama manusia, khususnya tim surveyor, dalam mempercepat penilaian tingkat kerusakan bangunan pascabencana. Dengan demikian, proses rehabilitasi dan rekonstruksi dapat dilakukan lebih efisien sehingga korban bencana dapat segera menerima bantuan. QS. Al-Maidah ayat 2 menegaskan pentingnya tolong-menolong dalam kebaikan:

وَتَعَاوَنُوا عَلَى الْبِرِّ وَالتَّقْوَىٰ وَلَا تَعَاوَنُوا عَلَى الْإِثْمِ وَالْعُدْوَانِ ۚ وَاتَّقُوا اللَّهَ ۚ إِنَّ اللَّهَ شَدِيدُ الْعِقَابِ

"Dan tolong menolonglah kamu dalam mengerjakan kebajikan dan takwa, dan jangan tolong-menolong dalam perbuatan dosa dan permusuhan. Bertakwalah kepada Allah, sesungguhnya Allah sangat berat siksaan-Nya." (Q.S. Al-Maidah: 2).

Ayat ini menegaskan bahwa tolong-menolong dalam kebaikan adalah perintah Allah yang harus dipatuhi. Dalam konteks ini, pengembangan sistem untuk mempercepat evaluasi kerusakan bangunan pascabencana merupakan implementasi nyata dari ajaran Islam dalam membantu sesama di situasi yang membutuhkan.

4.4.3 *Muamalah Ma'a Al-Alam*

Muamalah manusia dengan alam, atau hubungan manusia dengan alam, adalah elemen penting dalam Islam, seperti yang dinyatakan dalam ayat 41 surah Ar-Rum, yang menekankan betapa pentingnya bagi manusia untuk menjaga dan melestarikan alam, yang berbunyi:

ظَهَرَ الْفَسَادُ فِي الْبَرِّ وَالْبَحْرِ بِمَا كَسَبَتْ أَيْدِي النَّاسِ لِيُذِيقَهُمْ بَعْضَ الَّذِي عَمِلُوا لَعَلَّهُمْ يَرْجِعُونَ

"Telah nampak kerusakan di darat dan di laut disebabkan karena perbuatan tangan manusia, supaya Allah merasakan kepada mereka sebagian dari (akibat) perbuatan mereka, agar mereka kembali (ke jalan yang benar)." (Q.S. Ar-Rum: 41).

Ayat ini mengingatkan kita bahwa tindakan manusia yang melanggar keseimbangan alam menyebabkan kerusakan di Bumi. Bencana seperti banjir, tanah longsor, dan kebakaran hutan sering kali merupakan dampak dari perilaku manusia yang tidak bijaksana. Dalam konteks penelitian ini, teknologi yang dikembangkan bertujuan tidak hanya untuk menilai kerusakan, tetapi juga untuk membantu manusia lebih bertanggung jawab dalam menjaga lingkungan dan mencegah kerusakan lebih lanjut. Hal ini sejalan dengan prinsip Islam yang menyerukan kesadaran kolektif dalam memperbaiki hubungan dengan alam.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Penelitian ini berhasil mengembangkan Decision Support System (DSS) untuk menentukan tingkat kerusakan bangunan pasca-bencana alam dengan menggunakan metode *Local Binary Pattern* (LBP) dan *Principal Component Analysis* (PCA). Sistem ini mampu mengelompokkan gambar kerusakan bangunan menjadi tiga kategori: rusak ringan, rusak sedang, dan rusak berat. Proses dimulai dengan pengolahan gambar, diikuti dengan ekstraksi fitur menggunakan LBP, dan reduksi dimensi data menggunakan PCA. Sistem ini terbukti efektif dalam memberikan hasil yang relevan dengan kondisi lapangan, sehingga dapat menjadi alat yang andal untuk mendukung pengambilan keputusan secara cepat dalam situasi bencana.

5.2 Saran

Berdasarkan hasil penelitian ini, ada beberapa saran untuk pengembangan lebih lanjut:

1. Penelitian selanjutnya dapat memanfaatkan algoritma deep learning atau machine learning untuk meningkatkan akurasi clustering kerusakan bangunan. Model dapat dilatih untuk mengenali pola tekstur secara otomatis, sehingga tidak memerlukan ekstraksi fitur manual, serta divalidasi dengan dataset yang lebih beragam untuk memastikan generalisasi yang lebih baik.

2. Penggunaan Data Lebih Banyak: Penelitian ini menggunakan hanya 12 gambar dari BPBD Kota Malang. Untuk memperbaiki generalisasi sistem, perlu dilakukan pengujian dengan dataset yang lebih besar dan bervariasi, baik dari lokasi berbeda maupun kondisi kerusakan yang lebih kompleks.
3. Menambahkan metode perhitungan jarak: Untuk membandingkan titik-titik data yang memiliki kesamaan. Perhitungan jarak antar fitur dapat dilakukan secara lebih presisi menggunakan algoritma tertentu, sehingga mampu meningkatkan akurasi dalam membedakan tingkat kerusakan yang serupa.

Dengan saran-saran ini, diharapkan sistem yang dikembangkan dapat lebih efektif dalam membantu penilaian kerusakan bangunan pasca-bencana dan memberikan kontribusi signifikan dalam mitigasi bencana di masa depan.

DAFTAR PUSTAKA

- Abdi, M. F., Kurniawan, M. P., & others. (2022). *Local Binary Pattern* Untuk Ekstraksi Tekstur Gambar Wajah Menggunakan Masker Dan Tanpa Masker. *Technologia: Jurnal Ilmiah*, 13(2), 108–114.
- Achsani, F. N., Atmaja, R. D., & Purnamasari, R. (2015). Deteksi Adanya Cacat Pada Kayu Menggunakan Metode *Local Binary Pattern* . *eProceedings of Engineering*, 2(1).
- Almais, A. T. W., & Fatchurrochman, F. (2020). Implementasi fuzzy *Weighted product* penyusunan aksi rehabilitasi rekonstruksi pasca bencana berbasis decision support system dynamic. *ELTEK*, 18(1), 1–8.
- Almais, A. T. W., Sarosa, M., & Muslim, M. A. (2016). Implementation of multi experts multi criteria decision making for rehabilitation and reconstruction action after a disaster. *MATICS: Jurnal Ilmu Komputer Dan Teknologi Informasi (Journal of Computer Science and Information Technology)*, 8(1), 27–31.
- Almais, A. T. W., Susilo, A., Naba, A., Sarosa, M., Crysdiyan, C., Tazi, I., Hariyadi, M. A., Muslim, M. A., Basid, P. M. N. S. A., Arif, Y. M., & others. (2023). *Principal Component Analysis-Based Data Clustering* for Labeling of Level Damage Sector in Post-Natural Disasters. *IEEE Access*, 11, 74590–74601.
- Almais, A. T. W., Susilo, A., Naba, A., Sarosa, M., Juwono, A. M., Crysdiyan, C., Muslim, M. A., & Wicaksono, H. (2024). Characterization of Structural Building Damage in Post-Disaster using GLCM-PCA Analysis Integration. *IEEE Access*, 1–1. <https://doi.org/10.1109/ACCESS.2024.3469637>
- Alqoroni, S. (2023). *Implementasi COPRAS untuk menentukan tingkat kerusakan sektor pasca bencana alam*. Universitas Islam Negeri Maulana Malik Ibrahim.
- Ardiyanti, D. A. (2023). *Bantuan United States Agency For International Development (Usaid) Dalam Menangani Pandemi Covid-19 Di Filipina* [Thesis]. Universitas Pembangunan Nasional "Veteran" Yogyakarta.
- Arhandi, P. P., Mentari, M., & Romadhon, F. (2021). Kombinasi Metode Logical Binary Pattern dan K-Nearest Neighbor untuk Identifikasi Lubang pada Jalan Aspal. *Jurnal Nasional Pendidikan Teknik Informatika: JANAPATI*, 10(1), 11–22.
- Asmara, R., S.kom, & M.kom. (2017). *Sistem Informasi Pengolahan Data Penanggulangan Bencana Pada Kantor Badan Penanggulangan Bencana Daerah (BPBD) Kabupaten Padang Pariaman*. <https://api.semanticscholar.org/CorpusID:67735728>

- Bovik, A. C. (2009). Basic Binary Image Processing. In *The Essential Guide to Image Processing* (pp. 69–96). Elsevier. <https://doi.org/10.1016/B978-0-12-374457-9.00004-4>
- Costaner, L., Lisnawita, L., Guntoro, G., & Abdullah, A. (2024). Feature Extraction Analysis for Diabetic Retinopathy Detection Using Machine Learning Techniques. *Sistemasi: Jurnal Sistem Informasi*, 13(5), 2268–2276.
- Fahrurozi, A., Nugraha, N., & Rimirasih, D. (2023). *Sistem Klasifikasi Kayu Berbasis Citra Tekstur Menggunakan Machine Learning*. uwais inspirasi indonesia.
- Fuadah, A. F. (2023). *Klasifikasi tingkat kerusakan sektor pasca bencana alam menggunakan metode multimoora berbasis web* [PhD Thesis]. Universitas Islam Maulana Malik Ibrahim.
- Haryadi, H. (2009). *Administrasi perkantoran untuk manajer & staf*. VisiMedia.
- Hayqal, H. H. Q., Soesanto, O., & Sukmawaty, Y. (2022). K-Means Clustering dan Principal Component Analysis (PCA) Dalam Radial Basis Function Neural Network (RBFNN) Untuk Klasifikasi Data Multivariat. *Journal of Mathematics: Theory and Applications*, 1–7.
- He, C., Zhang, Q., Qu, T., Wang, D., & Liao, M. (2019). Remote Sensing and Texture Image Classification Network Based on Deep Learning Integrated with Binary Coding and Sinkhorn Distance. *Remote Sensing*, 11, 2870. <https://doi.org/10.3390/rs11232870>
- Humairah, A. A., Armansyah, Rasyidi, D. S., Pudza, I. W., Iwagumi, L. Z. A., Putri, N. W. E. U., Putra, R. R., Hardianti, S., & Pradjoko, E. (2023). Kewaspadaan Dan Penanggulangan Terhadap Ancaman Bencana Alam Di Desa Sembalun Kecamatan Sembalun Kabupaten Lombok Timur. *Portal ABDIMAS*, 1(2), 82–89. <https://doi.org/10.29303/portallabdimas.v1i2.3236>
- Kosasih, R., & Daomara, C. (2021). Pengenalan Wajah dengan Menggunakan Metode *Local Binary Pattern* s Histograms (LBPH). *Jurnal Media Informatika Budidarma*, 5(4), 1258–1264.
- Le, V. N. T., Ahderom, S., Apopei, B., & Alameh, K. (2020). A novel method for detecting morphologically similar crops and weeds based on the combination of contour masks and filtered *Local Binary Pattern* operators. *GigaScience*, 9(3), gaaa017.
- Prihantara, D., Oktaviani, M., & Iqbal Sj, M. (2023). Implementasi Decision Support System di Bidang Data Spasial: Systematic Literature Review. *Jurnal Sosial Sains*, 3(3), 307–314. <https://doi.org/10.36418/jurnalsosains.v3i3.712>
- Rahmaniati, A. F. (2022). *Penentuan tingkat kerusakan sektor pasca bencana alam menggunakan metode topsiis berbasis Machine Learning* [PhD Thesis]. Universitas Islam Negeri Maulana Malik Ibrahim.

- Rana Puspita, P. (2023). *Perancangan Sistem Informasi Perpustakaan Berbasis Web Pada MTs Darussalam Muara Tembesi* [Skripsi, UNAMA]. <http://repository.unama.ac.id/3583/>
- Rezty Amalia Aras. (2022). Decision Support System (DSS) dengan Berorientasi - Solver. *Jurnal Teknik Mesin, Elektro Dan Ilmu Komputer*, 2(1), 58–63. <https://doi.org/10.55606/teknik.v2i1.917>
- Ruswandi, R., Saefuddin, A., Mangkuprawira, S., Riani, E., & Kardono, P. (2008). Identifikasi Potensi Bencana Alam dan Upaya Mitigasi yang Paling Sesuai Diterapkan di Pesisir Indramayu dan Ciamis. *Jurnal Riset Geologi Dan Pertambangan*, 18(2), 1. <https://doi.org/10.14203/risetgeotam2008.v18.12>
- Safitri, A. H., Almais, A. T. W., Melani, R. I., & others. (2022). Pengujian optimization dan non-optimization query metode TOPSIS untuk penentuan tingkat kerusakan sektor bencana alam. *Jurna ELTIKOM*, 6(1), 89–99.
- Sudipa, I. G. I., Kharisma, L. P. I., Waas, D. V., Sari, F., Sutoyo, M. N., Rusliyadi, M., Setiawan, I., Martaseli, E., Sandhiyasa, I. M. S., Sulistianto, S., & others. (2023). *Penerapan Decision Support System (Dss) Dalam Berbagai Bidang (Revolusi Industri 4.0 Menuju Era Society 5.0)*. PT. Sonpedia Publishing Indonesia.
- Wenzel, H. (2009). *Health Monitoring of Bridges* (1st ed.). Wiley. <https://doi.org/10.1002/9780470740170>
- Yalesrie, D. N., Maharani, Y. N., Prasetya, J. D., & Cahyadi, T. A. (2023). Pengembangan Model Desain Rumah Bantuan bagi Keluarga Terdampak Bencana Angin Siklon di Indonesia. *Prosiding Seminar Nasional Teknik Lingkungan Kebumihan SATU BUMI*, 4(1). <https://doi.org/10.31315/psb.v4i1.8904>