

# Pengembangan *Deep Learning* Web Untuk Mendeteksi Video *Deepfake* Berbasis *Django Framework*

Lukas Yahya Adi Puspo<sup>#1</sup>, Herry Imanta Sitepu<sup>\*2</sup>, Dina Angela<sup>\*3</sup>

<sup>#</sup>Program Studi Teknik Komputer, Institut Teknologi Harapan Bangsa  
Jl. Dipatiukur no 80-84, Bandung, Jawa Barat, Indonesia

<sup>1</sup>lyadipuspo@gmail.com

<sup>2</sup>hisitepu@gmail.com

<sup>3</sup>dina\_angela@ithb.ac.id

*Abstract - In recent years, deepfakes have become an artificial intelligence technology that is growing very rapidly. thousands of millions of deepfake videos circulating on the internet widely. However, deepfakes still leave a negative trail and become a technology that is viewed negatively by the public. Millions of hoax videos, politics of fighting sheep to public pornography have been watched and distributed. So that many parties are harmed by the fake video. Many impacts resulting from the fake video attack and affect the lives of its victims. Therefore we need an application to detect a deepfake video so that it can become a victim to prove the video itself. This research was built to create a web application with deep learning based on convolutional neural network as a video facial feature extractor and long short term memory in classifying the video that will be requested. This web application called keapsuan.ai will make it easier for people from all walks of life to detect deepfake videos quickly and easily. With an accuracy percentage above 85% and a detection speed of only 1 minute, this web application has a pretty good performance to use. Also, this web application only requires three steps, uploading the video, detecting the video and downloading the result from the video. So that it can simplify the proof of deepfake videos that are widespread in the social-community..*

**Keywords:** *deepfake, hoax, website, deep learning, cnn.*

**Abstrak—** Dalam beberapa tahun terakhir, deepfake menjadi sebuah teknologi kecerdasan buatan yang peningkatannya sangat melonjak. Ribuan bahkan jutaan video deepfake beredar di internet secara luas. Namun deepfake tetap meninggalkan jejak negatif dan menjadi teknologi yang dipandang secara negative oleh masyarakat. Jutaan video hoax, politik adu domba sampai pornografi sudah masyarakat saksikan dan sebar. Sehingga banyak pihak yang dirugikan oleh kehadiran video palsu tersebut. Banyak dampak yang dihasilkan dari penyerangan video palsu tersebut dan mempengaruhi kehidupan korbannya. Maka dari itu diperlukan sebuah aplikasi untuk melakukan pendeteksian sebuah video deepfake agar dapat membantu korban untuk membuktikan video tersebut bukanlah dirinya. Penelitian ini dibangun untuk membuat aplikasi web dengan machine learning berbasis convolutional neural network sebagai peng-ekstrak fitur wajah video serta long short term memory dalam pengklasifikasian video yang akan dideteksi. Aplikasi web dengan nama keapsuan.ai ini akan memudahkan seluruh masyarakat dari berbagai kalangan untuk melakukan pendeteksian video deepfake dengan singkat dan mudah. Dengan

persentase keakuratan diatas 85% serta kecepeatan pendeteksian hanya 1 menit, membuat aplikasi web ini memiliki performa yang cukup baik untuk digunakan. Selain itu aplikasi web ini hanya memerlukan tiga langkah, mengunggah video, mendeteksi video dan mengunduh hasil dari video. Sehingga dapat menyederhanakan pembuktian video deepfake yang tersebar luas di masyarakat.

**Kata kunci:** *deepfake, hoax, website, deep learning, cnn*

## I. PENDAHULUAN

*Deepfake* adalah sebuah bentuk kecerdasan buatan yang digunakan untuk memanipulasi gambar atau video dalam sebuah peristiwa. Manipulasi yang dilakukan oleh *deepfake* adalah dengan menggabungkan anggota tubuh seseorang dengan wajah seorang yang lain maupun sebaliknya. *Deepfake* menggunakan teknologi kecerdasan buatan yang dikenal dengan *Generative Adversarial Network (GAN)*[1]. Namun *deepfake* sendiri tidak digunakan dengan baik oleh para penggunanya. Hampir 96% video *deepfake* yang beredar di internet mengandung muatan negatif. Muatan negative ini pun bermacam-macam bentuknya, politik adu domba, *hoax* yang seolah-olah disebarkan oleh figure ternama, bahkan sampai ke ranah pornografi [2]. Sehingga secara tidak langsung banyak pihak yang dirugikan oleh hal tersebut. Selain itu, perkembangan *deepfake* yang semakin merajalela juga dipengaruhi oleh kehadiran perangkat lunak yang semakin banyak. Baik yang digunakan oleh professional bahkan yang dapat masyarakat pada umumnya gunakan, seperti *FaceApp*, *faceSwap*, dll. Namun kehadiran perangkat pembuatnya tidak seimbang dengan perangkat lunak untuk mendeteksi video *deepfake* itu sendiri [3]. Oleh karena itu, perkembangan dari teknologi *deepfake* ini sendiri perlu diseimbangkan, agar tidak dianggap menjadi teknologi yang negatif.

Dari banyaknya dampak negatif yang disebabkan oleh *deepfake* itu sendiri, terutama video-video yang banyak tersebar merupakan video dengan wajah atau badan dari orang yang tidak diketahui tentunya banyak pihak yang akan merasa terfitnah oleh video yang mungkin bahkan mereka tidak pernah berpikir untuk membuatnya. Dampak yang akan ditinggalkan ke korban sendiri tentunya akan berdampak luas

## II. METODOLOGI

terhadap kehidupan pribadinya, selain dari merasa tidak nyaman di depan umum, korban juga dapat semakin tertekan apabila tidak dapat membuktikan video *deepfake* tersebut bukanlah dirinya [4].

Berdasarkan permasalahan tersebut, sudah ada beberapa solusi terkait dengan latar belakang yang sama untuk menemukan cara pendeteksian video *deepfake* tersebut dengan metode yang dapat digunakan masyarakat umum. *Kaggle.com* sebuah *website* dataset paling komplit di dunia, mengadakan sebuah lomba yang berlangsung selama 2 tahun yaitu *Kaggle Deepfake Detection Challenge* untuk menemukan teknologi paling efektif dalam mendeteksi manipulasi dalam video *deepfake* [5]. Selain itu riset yang dilakukan oleh *Springer Nature Singapore* juga menghasilkan metode *machine learning* dengan menggunakan *Key Frame Extraction* pada video yang diperoleh dari sosial media [6]. Tak hanya itu, *Konkuk University* di Korea juga menciptakan sebuah metode yaitu, *DeepVision*. Pendeteksian video *deepfake* dengan melihat pola kedipan mata pada subjek video. Karena salah satu kelemahan dari video *deepfake* itu sendiri adalah pola kedipan mata yang biasanya berubah dan tidak seperti seharusnya. Sehingga pola tersebut akan disesuaikan dengan pola kedipan mata yang seharusnya berdasarkan faktor umur dan jenis kelamin [7].

Dari solusi yang sudah ada, nyatanya masih belum terlalu membantu masyarakat umum untuk melakukan pembuktian *deepfake* karena kebanyakan masih berupa pemodelan *machine learning* yang tentu hanya orang-orang tertentu yang dapat menggunakannya. Sehingga masyarakat umum tetap kesulitan dalam mendeteksi video *deepfake* itu sendiri. Oleh karena itu, dengan membangun sebuah aplikasi *website* yang dapat diakses semua kalangan masyarakat dapat sangat membantu korban untuk membuktikan bahwa subjek dalam video bukanlah diri korban.

Salah satu solusi yang dapat mengatasi masalah tersebut adalah pembangun aplikasi *website* dengan penggunaan *machine learning* agar dapat mengetahui, membedakan dan memberi kesimpulan dari video yang ingin dibuktikan oleh masyarakat umum. Karena penggunaan *website* yang mudah, bisa dilakukan di berbagai perangkat tanpa melakukan unduhan apapun. Sehingga dapat mudah digunakan baik untuk pengguna pada umumnya yang hanya melakukan pendeteksian satu atau dua kali maupun pengguna yang memiliki perhatian khusus pada bidang video *deepfake* yang akan melakukan pendeteksian berulang.

Maka dari itu, hasil akhir dari penelitian ini adalah rekayasa perangkat lunak berupa aplikasi web yang dapat digunakan masyarakat pada umumnya untuk melakukan pendeteksian video *deepfake*. Penelitian ini sendiri akan menggunakan metode *Convolutional Neural Network* dan *Long Short Term Memory*. Untuk melakukan pemisahan dan klasifikasi dari video yang akan dilatih.

Aplikasi web ini sendiri diharapkan dapat membantu para korban video *deepfake* baik dari kalangan manapun untuk membuktikan bahwa subjek pada video tersebut bukan merupakan dirinya. serta dapat digunakan oleh pengguna umum sebagai pemeriksa keaslian sebuah video.

Dalam perancangan dan pengembangan sistem *monitoring* lansia, perlu dilakukan peninjauan pustaka sebagai dasar dari perancangan dan pengembangan sistem. Tinjauan pustaka akan dilakukan dengan mengidentifikasi masalah, menganalisis sistem yang sudah ada, sistem yang diusulkan, dan analisis kebutuhan sistem.

### A. Identifikasi Masalah

Sebuah video *deepfake* yang tersebar luas, biasanya sudah dibentuk sedemikian rupa agar menyerupai wujud asli dari subjeknya. Sehingga hal tersebut tidak dapat diketahui dengan telanjang mata apalagi oleh masyarakat pada umumnya. Karena untuk membuat sebuah video *deepfake* dengan tingkat realitas yang hampir menyerupai aslinya, dibutuhkan seseorang yang mengerti dengan mekanisme dari *machine learning* itu sendiri. Maka dari itu, kebanyakan dari video *deepfake* dengan muatan negatif jauh lebih halus daripada video *deepfake* yang beredar di social media dengan menggunakan aplikasi yang dapat diunduh secara bebas.

Selain itu, jumlah perangkat lunak yang untuk membuat video *deepfake* semakin mudah dicari dan digunakan, serta beredar luas di masyarakat. Baik dalam bentuk aplikasi, maupun metode *machine learning* yang dapat diimplementasikan. Namun jumlah itu justru berbanding terbalik dengan perangkat lunak untuk melakukan pendeteksian sebuah video *deepfake* apakah video tersebut asli atau palsu. Tidak terlalu banyak tools yang tersedia di internet untuk melakukan deteksi video *deepfake* yang dapat digunakan oleh masyarakat pada umumnya. Dan kebanyakan tools tersebut juga berupa pemodelan *machine learning* bukan sebuah aplikasi yang sudah jadi secara utuh dan dapat digunakan.

Oleh karena itu, untuk menghadapi permasalahan-permasalahan tersebut, penerapan aplikasi web tentu dapat memudahkan masyarakat dalam melakukan pendeteksian tersebut. Karena dengan web yang dapat dibuka oleh siapa saja, maka semua orang dapat melakukan deteksi video *deepfake* lagi tanpa kesulitan, sesuai dengan tujuan awal dari penelitian ini.

### B. Riset terkait

Terdapat beberapa jurnal yang mengangkat topik mengenai model yang dapat digunakan untuk mendeteksi video *deepfake*. Salah satunya adalah metode yang dibangun oleh *Springer Nature Singapore*, dimana mereka menggunakan metode *key video extraction* dalam pengambilan frame video yang akan dideteksi dimana hanya awal dan akhir adegan kunci saja yang di ambil oleh sistem. *Key frame* tadi akan diklasifikasi menggunakan model *The Xception* dan dapat menghasilkan akurasi sampai 92.33% dari 600 video yang diuji [6].

Selain itu *Konkuk University* mengembangkan produk mereka yaitu, *DeepVision* yang merupakan pendeteksi video *deepfake* menggunakan pola kedipan mata subyek video apakah sesuai dengan jumlah kedipan mata pada usia dan jenis kelamin subyek. Dari model yang dibangun tersebut

mereka berhasil mencapai akurasi 87.50 dengan mendeteksi tujuh dari delapan video yang diuji [7]. Dari kedua riset tersebut, peneliti dapat menyimpulkan beberapa perbandingan dari teknologi, kelebihan dan kekurangan masing-masing riset yang dijabarkan pada Tabel I.

Terdapat juga DetectFakes [8], sebuah *website* yang dibangun oleh Massachusetts *Institute of Technology* untuk membantu orang-orang membedakan video *deepfake*. Pada *website* tersebut disediakan soal-soal berupa video yang dapat ditebak oleh penggunanya apakah video tersebut asli atau palsu seperti pada Gambar 1, salah satu contoh soal yang ditampilkan oleh *Detect Fakes* tersebut.

### C. Sistem yang Sudah Ada dan Sistem yang Diusulkan

*Springer Nature Singapore* menciptakan sebuah model *deep learning* untuk melakukan pendeteksian dengan metode *Convolutional Neural Network* dengan metode *pre-processing* *Key Frame Extraction*, yang dapat dilihat lebih lengkap pada Gambar 2.

Dengan metode *Key Frame Extraction* tersebut, memudahkan model dalam mengambil frame video yang akan dijadikan diklasifikasikan. *Key Frame Extraction* adalah pengambilan frame video hanya pada awal dan akhir sebuah adegan atau aksi pada sebuah video. Lalu frame tersebut akan melalui *face detection* dimana video akan dipusatkan ke muka dan di *crop and resized*.

Setelah itu model akan menggunakan *Convolutional Neural Network* sebagai pelabelan dan hasilnya akan diklasifikasi menggunakan *The Xception Model* untuk diperoleh hasil apakah video tersebut merupakan video asli atau palsu. Selain itu pada akhir penelitian mereka melakukan perbandingan dengan menggunakan metode klasifikasi yang lain yaitu Resnet50 dan InceptionV3. Hasil akurasi yang diperoleh sebesar 92.33% dengan menggunakan Xception dan menjadi hasil paling tinggi dibanding lainnya

TABEL I

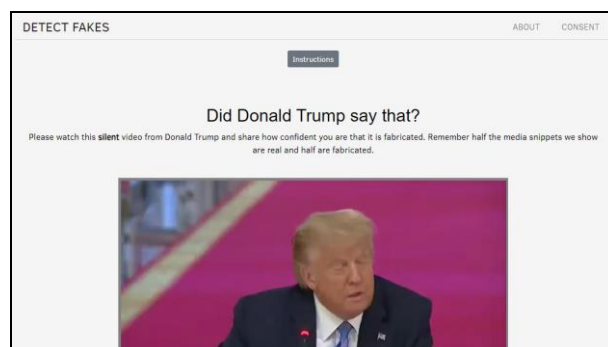
PERBANDINGAN METODE PADA RISET TERKAIT

Riset	Teknologi	Kelebihan	Kekurangan
Metode pendeteksian video <i>deepfake</i> menggunakan <i>key frame</i> dan CNN Model	Menggunakan <i>key frame extraction</i> dalam ekstraksi fitur dan klasifikasi <i>the Xception</i>	Penggunaan <i>key frame extraction</i> membuat kinerja CNN tidak terlalu berat karena <i>frame</i> diambil secara efektif	Metode <i>The Xception</i> sulit untuk diterapkan
Metode pendeteksian video <i>deepfake</i> dengan menggunakan pola kedipan mata subjek	Menggunakan <i>Fast-Hyperface</i> untuk mendeteksi pola kedipan mata berdasarkan <i>frame</i> pada video	Metode yang digunakan berbeda dari umumnya dan didasarkan pada pola bagian tubuh bukan pola wajah	Metode yang diterapkan terlalu sulit dan diperlukan perhitungan yang sangat akurat untuk menemukan pola yang sesuai

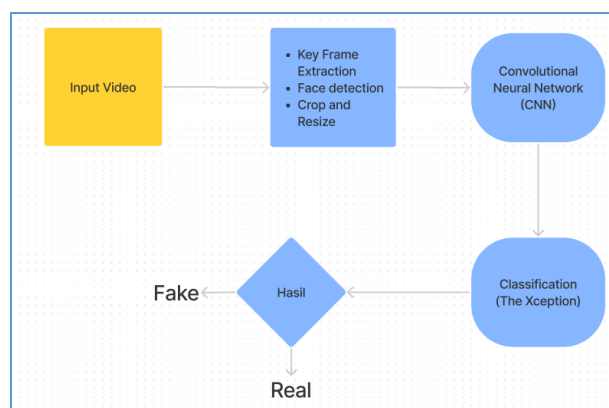
Terdapat dua alur utama pada system yang dibangun, yaitu Alur prediksi dan Alur *Training*. Alur prediksi merupakan alur yang akan dilalui oleh pengguna aplikasi web tersebut. Sementara alur training merupakan alur dari sistem pemodelan *machine learning* yang akan digunakan peneliti sebagai *trained model* yang akan diletakan pada *backend website*.

Pada alur *training*, *dataset* yang dibagi menjadi dua yaitu *data testing* dan *data training* sebanyak 80 banding 20. Lalu *dataset* tersebut akan melalui proses *pre-processing* dengan memecah video menjadi *frame by frame* lalu melakukan pendeteksian wajah dan melakukan *crop* pada wajah tersebut agar pada saat diekstrak, fitur yang dilakukan jelas yaitu wajah subjek video. Setelah itu fitur wajah tersebut akan diekstrak oleh *Convolutional neural Network* dan akan diklasifikasi menggunakan metode *Long Short Term Memory* untuk dipisahkan yang mana video asli dan yang mana video *deepfake*. Hasil dari *Trained Model* tersebut yang akan dimuat pada aplikasi web.

Sementara alur prediksi adalah alur yang akan dilalui oleh video unggahan pengguna pada *website*. Video yang diunggah akan melewati proses *pre-processing* juga. Sama seperti video pada alur *training*. Setelah itu, video akan diklasifikasi menggunakan model yang sudah dilatih sebelumnya



Gambar 1 *Detect Fakes*



Gambar 2 *Flowchart* sistem yang sudah ada

untuk menghasilkan kesimpulan apakah video tersebut merupakan video asli atau palsu. Pengguna akan memperoleh hasil *output* berupa keputusan serta persentase dari keputusan video tersebut.

Alur-alur yang dijelaskan di atas, merupakan penjabaran dari *flowchart* system yang telah peneliti susun. Untuk melihat *flowchart* tersebut secara lengkap dapat dilihat pada Gambar 3.

#### D. Analisis Kebutuhan

Pada bagian kebutuhan, pengembangan sistem akan dibagi menjadi dua bagian, yaitu kebutuhan fungsional dan kebutuhan non fungsional. Kebutuhan fungsional merupakan kebutuhan yang berisi proses yang akan dilakukan sistem, sedangkan kebutuhan non fungsional merupakan spesifikasi yang dibutuhkan oleh sistem.

Kebutuhan fungsional merupakan semua fungsi atau layanan yang diberikan oleh sistem kepada pengguna sistem yang mencakup respon sistem terhadap masukan dari pihak pengguna. Kebutuhan Fungsional ini dapat dilihat di Tabel II.

Sementara kebutuhan non-fungsional adalah kebutuhan yang dimiliki sistem secara spesifik dan dapat mengalami pengembangan. Kebutuhan non-fungsional dapat dilihat di Tabel III.

TABEL II  
KEBUTUHAN NON-FUNGSIONAL

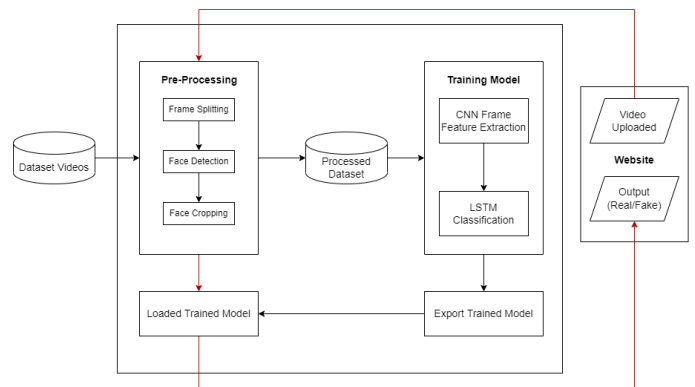
Kode	Kebutuhan Fungsional
F001	Pengguna dapat membuka aplikasi web dengan komponen – komponen yang lengkap
F002	Pengguna dapat melakukan pencarian berkas video melalui tombol yang disediakan
F003	Pengguna dapat melakukan pengunggahan video dan dapat menampilkan video ke dalam aplikasi web
F004	Pengguna dapat melihat proses yang sedang dilakukan sistem terhadap video unggahannya
F005	Sistem dapat mendeteksi wajah objek di dalam video unggahan pengguna
F006	Video unggahan pengguna dapat dipecah <i>frame per frame</i> lalu diubah ke dalam bentuk gambar
F007	Pengguna akan memperoleh hasil dengan waktu prediksi sekitar 1 menit dengan akurasi diatas 60%
F008	Pengguna dapat melihat hasil akhir pendeteksian beserta bukti-bukti yang mendukung
F009	Pengguna dapat melakukan unduhan dari hasil prediksi yang telah diperoleh dalam bentuk .pdf

#### E. Arsitektur Sistem

Dalam perancangan sistem secara keseluruhan, seperti yang telah disebutkan sebelumnya, akan terbagi ke dalam dua subsistem yaitu; subsistem *training* yang akan membahas pelatihan model *machine learning* hingga dapat digunakan ke dalam aplikasi web. Dan subsistem prediksi yang akan berfokus dalam sistem aplikasi web yang akan digunakan untuk pengguna dalam melakukan pendeteksian video.

Pada Subsistem *training* akan menjelaskan bagaimana sistem dapat mengetahui sebuah wajah dalam video beserta pemecahan video frame per frame sebelum video tersebut diproses lebih lanjut oleh *convolutional neural network* untuk mengekstraksi fitur wajah dan dilabeli oleh *long short term memory* apakah fitur-fitur pada wajah tersebut merupakan fitur yang termasuk video asli atau video *deepfake*. Hasil akhir dari Subsistem ini merupakan model *pytorch* yang akan digunakan di dalam aplikasi web.

Lalu pada Subsistem prediksi akan menjelaskan arsitektur sistem dari aplikasi web pada penelitian ini yang menggunakan *framework* Django berbasis python. Beserta halaman-halaman serta urutan proses yang akan dilalui oleh pengguna dan juga video unggahan oleh pengguna tersebut. Dimana proses yang serupa dengan subsistem *training* akan dilakukan kembali ke dalam subsistem prediksi mulai dari tahapan *preprocessing* hingga ke dalam tahapan prediksi.



Gambar 3 Flowchart sistem yang diusulkan

TABEL III  
KEBUTUHAN NON-FUNGSIONAL

Kode	Kebutuhan non-Fungsional
NF001	Sistem dapat menjalankan <i>pre-trained model</i> sesuai dengan batas-batas serta spesifikasi kemampuan aplikasi web
NF002	<i>Pre-trained model</i> dapat menghasilkan akurasi prediksi diatas 70%

### F. Subsistem training

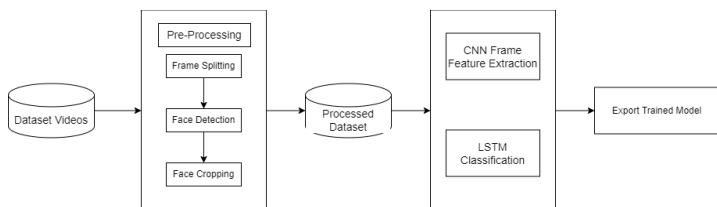
Subsistem training merupakan tahapan awal perancangan sistem aplikasi web. Dimana pada subsistem ini, akan dilakukan pelatihan algoritma *deep learning* untuk dapat melakukan pendeteksian serta memahami cara membedakan video asli dan video palsu. Subsistem *training* ini merupakan kegiatan melakukan pembangunan ulang model, yang sesuai dengan spesifikasi penelitian ini dan telah disusun berdasarkan algoritma yang telah disusun pada sumber [9]. Pada algoritma tersebut memiliki tiga komponen utama, yaitu *Dataset Collection*, *Preprocessing* dan *Model Training*. Urutan dari Subsistem *training* dapat dilihat pada Gambar 4.

*Dataset Collection* merupakan tahapan awal sebelum melakukan proses pelatihan. Dimana mengumpulkan data-data yang relevan dengan penelitian dan akan digunakan dalam pelatihan model. Dengan jumlah dataset total sebanyak 5870 video lalu dataset tersebut akan dibagi menjadi 80:20 sebagai data *training* dan data validasi.

Lalu masuk ke tahap *Pre-processing* yaitu proses dimana video-video dalam dataset tersebut disederhanakan bentuknya menjadi bentuk yang diperlukan dalam penelitian ini saja. Dimana bagian utama yang digunakan merupakan area wajah pada objek saja. Dalam *peprocessing* sendiri terdapat tiga tahapan yaitu; *frame splitting* untuk melakukan pembagian *frame-frame* dalam video, *face detection* untuk melakukan pendeteksian wajah objek di dalam video dan terakhir adalah *face cropping*, dimana video akan dipotong dan dipusatkan hanya ke bagian wajah dari objek saja.

Setelah itu, *dataset* yang telah di *pre-processed* tersebut kembali disimpan menjadi *processed dataset* yang akan digunakan dalam *model training*.

Terakhir, *processed dataset* tersebut sudah dapat menjadi bahan untuk pelatihan *machine learning* yang akan dipergunakan. *Training machine learning* ini sendiri menggunakan beberapa faktor yang dapat diatur seperti; *epochs*, *batch size*, *learning rate* dan *frame rate*. Pelatihan model *machine learning* ini memakai dua metode *deep learning* yaitu *Convolutional Neural Network* dan *Long Short Term Memory*. *CNN* disini akan menggunakan arsitektur *ResNext50* sebagai ekstraksi fitur dalam video lalu *LSTM* digunakan untuk melakukan pemrosesan berulang untuk melakukan klasifikasi terhadap video apakah video tersebut masuk ke dalam kelas video *REAL* atau video *FAKE*. Dari proses pelatihan tersebut maka akan mengeluarkan output akhir berupa model dalam format *pytorch (.pt)*, grafik akurasi beserta *loss* dan *confusion matrix*.



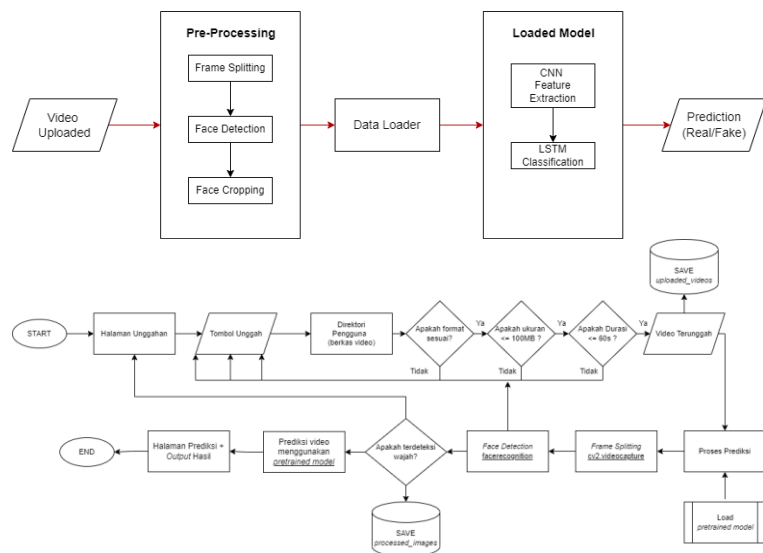
Gambar 4 Blok diagram subsistem *training*

### G. Subsistem prediksi

Subsistem prediksi merupakan subsistem yang akan membahas alur serta urutan sistem yang dilalui oleh pengguna aplikasi web. Aplikasi web yang dibangun akan menggunakan *framework* Django dengan Bahasa pemograman *python*. Selain itu juga tetap akan menggunakan dasar-dasar pembangunan web seperti *HTML*, *CSS* dan *Javascript* sebagai pendukung performa aplikasi web itu sendiri. Lalu, karena *pretrained-model* yang digunakan menggunakan *PyTorch (.pt)* maka pada arsitektur aplikasi web akan ditambahkan *Pytorch* agar dapat menjalankan *pretrained model* yang digunakan.

Pada dasarnya aplikasi web akan terbagi menjadi dua halaman; halaman unggahan dan halaman hasil. Halaman unggahan merupakan halawan awal untuk pengguna dapat melakukan pengunggahan video melalui *widget upload button* yang disediakan pada halaman tersebut. Video tersebut akan dilihat apakah memenuhi syarat dari pendeteksian video *deepfake* atau tidak. Jika sudah memenuhi semua syarat, lalu video akan diproses oleh sistem dengan melalui tahapan yang sama pada subsistem *training* sebelumnya. Mulai dari *preprocessing* hingga prediksi yang dilakukan oleh model dari subsistem *training*.

Setelah itu, apabila pendeteksian sudah selesai dilakukan oleh sistem, maka pengguna akan dialihkan ke halaman hasil, dimana halaman tersebut akan menampilkan hasil dari prediksi video beserta dengan tambahan-tambahan untuk menambah bukti dari hasil tersebut. Hasil prediksi tersebut nantinya dapat diunduh oleh pengguna dalam bentuk *Portable Document Format (.pdf)* sehingga pengguna tidak perlu melakukan pendeteksian berulang terhadap video yang sama. Untuk memperjelas alur kerja subsistem prediksi yang akan dibahas, dapat dilihat pada blok diagram serta flowchart di Gambar 5.



Gambar 5 Blok Diagram + *flowchart* subsistem prediksi

### III. HASIL DAN PEMBAHASAN

TABEL V  
SKENARIO PENGUJIAN FUNGSIONAL

#### A. Tujuan Pengujian

Tujuan dari penelitian ini sendiri adalah merancang dan mengembangkan sebuah aplikasi web yang dapat membantu masyarakat serta figur publik yang merasa dirugikan oleh kehadiran video *deepfake* untuk membuktikan bahwa subjek dalam video tersebut bukanlah dirinya. Agar dapat mencapai tujuan tersebut, akan dilakukan pengujian terhadap aspek kebutuhan fungsional sistem dalam aplikasi web. Pengujian ini akan dibagi-bagi berdasarkan subsistem yang telah dibahas pada bab sebelumnya. Pengujian ini akan diujicoba menggunakan skenario-skenario yang telah ditentukan hingga akhirnya dapat dipastikan bahwa seluruh sistem dapat menjalankan fungsinya dengan baik

#### B. Skenario Pengujian

Dalam pengujian sistem keseluruhan, scenario akan dibagi berdasarkan subsistem yang telah dijelaskan sebelumnya. Dimana subsistem *training* akan berfokus kepada performa prediksi dari model yang diperoleh, sementara subsistem prediksi akan berfokus terhadap kelengkapan, kelancaran serta ketepatan sistem aplikasi web saat digunakan oleh pengguna. Berdasarkan hal tersebut, terdapat beberapa scenario untuk skenario Non Fungsional pada Tabel IV dan scenario pengujian fungsional pada Tabel V.

#### C. Hasil Pengujian dan Analisa

Berdasarkan scenario yang sudah disusun sebelumnya, dihasilkan beberapa hasil yang dapat menjadi bahan analisa dan menjadi parameter untuk melihat apakah sistem aplikasi web sudah memenuhi kebutuhan fungsional atau belum.

TABEL IV  
SKENARIO PENGUJIAN NON-FUNGSIONAL

Kode	Tujuan	Skenario	Status
NF001	Sistem dapat menjalankan <i>pre-trained model</i> sesuai dengan batas-batas serta spesifikasi kemampuan aplikasi web	Pelatihan model menggunakan pengaturan standar pada aplikasi ( <i>sequence length, epochs, batch size</i> )	Berhasil
NF002	<i>Pre-trained model</i> dapat menghasilkan akurasi prediksi diatas 70%	Pengujian <i>pretrained model</i> di dalam aplikasi web	Berhasil

Kode	Tujuan	Skenario Pengujian	Status
F001	Pengguna dapat membuka aplikasi web dengan komponen – komponen yang lengkap	Mengecek apakah setiap halaman aplikasi web memunculkan komponen yang sudah ditentukan	Berhasil
F002	Pengguna dapat melakukan pencarian berkas video melalui tombol yang disediakan	Melalui tombol ‘pilih file’ dapat membuka direktori local pengguna dengan hanya menampilkan berkas berekstensi video	Berhasil
F003	Pengguna dapat melakukan pengunggahan video dan dapat menampilkan video ke dalam aplikasi web	Video pilihan pengguna dapat diunggah dan ditampilkan ke dalam aplikasi web	Berhasil
F004	Pengguna dapat melihat proses yang sedang dilakukan sistem terhadap video unggahannya	Saat proses pendeteksian dimulai, setiap perubahan statu pendeteksian dapat dimunculkan pada halaman pengguna	Berhasil
F005	Sistem dapat mendeteksi wajah objek di dalam video unggahan pengguna	Sistem dapat melakukan pendeteksian wajah objek, dan memfilter video yang tidak memiliki wajah di dalam video	Berhasil
F006	Video unggahan pengguna dapat dipecah <i>frame</i> per <i>frame</i> lalu diubah ke dalam bentuk gambar	Video unggahan pengguna dapat dipecah berdasarkan <i>sequence length model</i> lalu diubah ke bentuk gambar untuk ditampilkan ke pengguna	Berhasil
F007	Pengguna akan memperoleh hasil dengan waktu prediksi sekitar 1 menit dengan akurasi diatas 60%	Untuk video dengan ukuran 10 – 20 MB, aplikasi web dapat memprediksi dengan durasi 1 – 2 menit	Berhasil
F008	Pengguna dapat melihat hasil akhir pendeteksian beserta bukti-bukti yang mendukung	Pada halaman hasil prediksi, pengguna dapat melihat hasil akhir, persentase, <i>processed images</i> dan video unggahannya	Berhasil
F009	Pengguna dapat melakukan unduhan dari hasil prediksi yang telah diperoleh dalam bentuk .pdf	Sistem aplikasi web, dapat mengkonversi halaman hasil prediksi ke bentuk PDF dengan komponen yang sama dan lengkap dengan ukuran dibawah 5 MB	Berhasil

1) Hasil pengujian subsistem training

Melalui dua scenario yang telah ditentukan, terdapat dua pengujian yang dilakukan pada subsistem *training* ini, yang pertama untuk melakukan pelatihan sehingga diperoleh *pretrained model* dengan akurasi yang telah ditentukan, serta melakukan tes terhadap *pretrained model* ke dalam aplikasi web secara langsung untuk mengetahui performa dari model itu sendiri. Dalam pengujian ini, menggunakan dua parameter yang dilakukan perubahan yaitu *sequence length* dan *epochs*. Sehingga diperoleh 4 *pretrained model* dengan nilai akurasi dan nilai loss pada Tabel VI.

Setelah diperoleh hasil-hasil tersebut, maka keempat *pretrained model* diuji langsung ke dalam aplikasi web yang telah dibangun menggunakan 5 sampel video dengan kondisi yang berbeda-beda seperti pada Tabel VII. Dan dari *pretrained model* yang telah diperoleh dapat memberikan hasil akurasi dan durasi yang dijelaskan pada Tabel VIII. Melalui hasil tersebut, dapat diperoleh rata-rata performa setiap model berdasarkan akurasi dan durasi yang dapat menjadi acuan pengambilan simpulan dari performa model pada Tabel IX.

Melalui pengujian tersebut dapat diperoleh simpulan berdasarkan analisa antara lain:

- Model 20 *Sequence Length* - 10 *Epochs* karena memiliki peningkatan rata-rata akurasi sebesar 2,34% dengan penurunan rata-rata *loss* sebesar 0,1477 dibanding dengan model-model yang lain.

TABEL VI

HASIL AKURASI DAN LOSS DARI PELATIHAN MODEL

MODEL	Seq.length	Epochs	Avg. Akurasi		Avg. Loss	
			Avg. Akurasi Training	Avg. Akurasi Validasi	Avg. Loss Training	Avg. Loss Validasi
10	10		90,14	87,43	0,3480	0,3460
10	20		93,83	88,97	0,2868	0,3513
20	10		87,40	89,74	0,4855	0,3378
20	20		92,22	89,66	0,4348	0,2930

TABEL VII

SAMPel VIDEO PENGUJIAN

Video ID	Tangkapan Video	Durasi (s)	Ukuran (Mb)	Status Video
1		5	6,48	Video Asli
2		4	5,16	Video Asli
3		7	8,65	Video Asli

Video ID	Tangkapan Video	Durasi (s)	Ukuran (Mb)	Status Video
4		13	2,03	Video Palsu
5		10	9,84	Video Palsu

TABEL VIII

HASIL PENGUJIAN *PRETRAINED MODEL* MENGGUNAKAN SAMPEL VIDEO

Video ID	Model 10 <i>sequence length</i>		Model 20 <i>sequence length</i>	
	10 <i>epochs</i>	20 <i>epochs</i>	10 <i>epochs</i>	20 <i>epochs</i>
1	REAL	REAL	REAL	REAL
	Durasi: 40s	Durasi: 34s	Durasi: 79s	Durasi: 88s
	Akurasi: 78,9%	Akurasi: 74,5%	Akurasi: 73,9%	Akurasi: 80%
2	REAL	REAL	REAL	REAL
	Durasi: 32s	Durasi: 35s	Durasi: 67s	Durasi: 70s
	Akurasi: 98,9%	Akurasi: 99%	Akurasi: 89,6%	Akurasi: 98%
3	REAL	REAL	REAL	REAL
	Durasi: 34s	Durasi: 34s	Durasi: 67s	Durasi: 70s
	Akurasi: 96,5%	Akurasi: 90,9%	Akurasi: 89%	Akurasi: 98,7%
4	FAKE	FAKE	FAKE	FAKE
	Durasi: 19s	Durasi: 19s	Durasi: 34s	Durasi: 42s
	Akurasi: 70%	Akurasi: 93,6%	Akurasi: 67,1%	Akurasi: 67%
5	FAKE	FAKE	FAKE	FAKE
	Durasi: 39s	Durasi: 35s	Durasi: 63s	Durasi: 63s
	Akurasi: 95%	Akurasi: 78%	Akurasi: 72,5%	Akurasi: 72%

TABEL IX

TABEL RATA-RATA PERFORMA SETIAP MODEL DALAM APLIKASI WEB

Model	Seq. length	Epochs	Rata-rata Durasi	Rata-rata Akurasi
			Prediksi	Prediksi
10	10		32,8s	87,86%
10	20		31,4s	87,20%
20	10		62s	78,42%
20	20		66,6s	83,14%

- Dengan menggunakan *sequence length* yang lebih banyak tidak terlalu berarti dalam meningkatkan performa akurasi prediksi, model dengan *sequence length* = 10, memiliki akurasi hingga 87% dengan rentang durasi sekitar 30 detik. Sementara model 20 *sequence length*, hanya memperoleh akurasi sekitar 80,78% dengan durasi pendeteksian yang 2 kali lebih lama dibanding model 10 *sequence length* sekitar 60 detik.
- Terdapat dua model yang efektif digunakan di dalam aplikasi web, yaitu model dengan 10 *sequence length* baik model 10 *epochs* dan 20 *epochs*. Karena kedua model tersebut memiliki selisih sangat kecil baik dalam durasi dengan selisih 1.6 detik serta akurasi memiliki selisih 0,6%.

## 2) Hasil pengujian subsistem prediksi

Lalu pada subsistem prediksi, inti dari pengujian yang dilakukan adalah melihat apakah seluruh komponen aplikasi web dapat muncul atau tidak. Lalu video unggahan pengguna juga dapat diproses dengan baik oleh system aplikasi web dimulai dari saat pengguna membuka aplikasi web, pengguna memilih dan mengunggah video, lalu proses saat video akan dan saat dideteksi oleh system, hingga aplikasi web menampilkan hasil akhir dari video unggahan pengguna tersebut. Serta apakah pengguna dapat melakukan pengunduhan hasil prediksi terhadap video unggahannya dalam bentuk .pdf. Setiap hasil akhir komponen dapat dilihat dari 3 gambar yang terpisah yaitu; Gambar 6 Halaman unggahan, Gambar 7 Halaman hasil prediksi dan Gambar 8 hasil unduhan pdf.

Melalui pengujian seluruh sistem aplikasi web tersebut ada subsistem prediksi, dapat ditemukan simpulan antara lain:

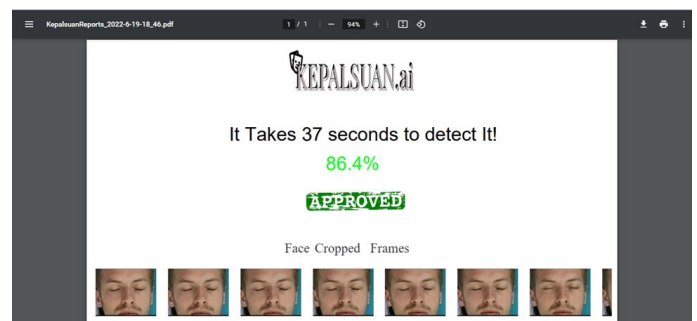
- Kecepatan prediksi dipengaruhi oleh ukuran video yang sedang diprediksi beserta berapa *frame* atau *sequence length* yang digunakan di dalam model. Sementara durasi video tidak mempengaruhi durasi dari prediksi itu sendiri. Video ID 4 memiliki durasi 12 detik sementara ukuran videonya hanya 2 MB, sehingga video tersebut dapat diprediksi dengan durasi yang paling singkat dibanding dengan video sampel lainnya.



Gambar 6 Tampilan Halaman Unggahan



Gambar 7 Tampilan Halaman Hasil Prediksi



Gambar 8 Tampilan berkas .pdf

## IV. SIMPULAN

Setelah melalui serangkaian proses dalam penelitian ini, dapat diketahui bahwa penelitian ini telah berhasil dalam melakukan pengembangan sebuah aplikasi web dengan *deep learning* berbasis *Django Framework* untuk dapat membedakan video dengan wajah tunggal apakah video tersebut merupakan video asli atau video palsu.



Dengan memiliki tingkat rata-rata akurasi diatas 85% serta durasi prediksi hanya dengan rentan waktu 1 – 2 menit, dan komponen-komponen serta alur yang sederhana menjadikan aplikasi web ini sudah menjadi aplikasi web yang mudah dipahami serta singkat dan cepat bagi kalangan umum yang sudah biasa melakukan aktifitas di internet. Sehingga aplikasi web ini dapat memenuhi tujuan dari penelitian ini yaitu untuk membangun media bagi masyarakat umum untuk membantu melakukan pendeteksian video *deepfake* yang tersebar di masyarakat sehingga dapat meminimalisir resiko untuk menjadi korban.

#### DAFTAR REFERENSI

- [1] Tirto.id. (30 Nov 2020) Apa itu Deepfake dan Bagaimana Cara kerjanya? [Daring]. Tersedia: <https://tirto.id/apa-itu-deepfake-dan-bagaimana-cara-kerjanya-f7to>. [Diakses: 13 Oktober 2021]
- [2] Wired. (7 Okt 2019) Most Deepfakes Are Porn, and They're Multiplying Fast [Daring]. Tersedia: <https://www.wired.com/story/most-deepfakes-porn-multiplying-fast/>. [Diakses: 13 Oktober 2021]
- [3] A. Henry, P. giorgio, C. Francesco, C. Laurence, *Deeptrace: The State of Deepfakes*. Amsterdam: Deeptrace labs, Sep 2019.
- [4] CBS News. (14 Okt 2021) Completely horrifying, dehumanizing, degrading: One woman's fight against deepfake porn [Daring]. Tersedia: <https://www.cbsnews.com/news/deepfake-porn-woman-fights-online-abuse-cbsn-originals/>. [Diakses: 14 Oktober 2021]
- [5] Kaggle. (24 Apr 2020). Kaggle's Deepfake Detection Challenge, [Daring]. Tersedia: <https://www.kaggle.com/c/deepfake-detection-challenge>. [Diakses: 13 Oktober 2021]
- [6] Mitra, A., Mohanty, S.P., Corcoran, P. et al. "A Machine Learning Based Approach for Deepfake Detection in Social Media Through Key Video Frame Extraction". SN COMPUT. SCI. 2, 98 (2021). [Daring]. Tersedia: <https://doi.org/10.1007/s42979-021-00495-x>. [14 Oktober 2021]
- [7] T. Jung, S. Kim and K. Kim, "DeepVision: Deepfakes Detection Using Human Eye Blinking Pattern", *IEEE Access*, vol. 8, pp. 83144-83154, February 14, 2020.
- [8] Mit.edu. Detect Fakes [Daring]. Tersedia: <https://detectfakes.media.mit.edu/>. [Diakses: 15 Oktober 2021]
- [9] A. Jadhav, A. Patange, J. Patel, H Patil, M Mahajan, "Deepfake Video Detection using Neural Networks", *International Journal for Scientific Research & Development*, Vol. 8, Issue 1, 2020.

**Lukas Yahya Adi Puspo**, kelahiran kota Bandung. Lulusan SMA Santa Maria 2 Bandung. Dan sedang menjalani pendidikan sarjana di Institut Teknologi Harapan Bangsa Bandung jurusan Media Internet Technology.

**Herry Imanta Sitepu**, menempuh pendidikan S1 di Teknik Elektro ITB dan lulus tahun 1999, dan memperoleh gelar magister dan doktor di jurusan yang sama di ITB. Sejak tahun 2006 aktif sebagai pengajar di Prodi Sistem Komputer ITHB. Minat penelitian: *computer networking, programming dan distributed system*.

**Dina Angela**, kelahiran Bandung 1974, menyelesaikan S1 Jurusan Teknik Elektro bidang Telekomunikasi di Universitas Kristen Maranatha pada 1999 dan S2 Jurusan Teknik Elektri bidang Telekomunikasi di Institut Teknologi Bandung pada 2003. Bidang penelitian antena dan propagasi dan sistem komunikasi.