

# Pemodelan Teori Graf pada Model Klasifikasi Gender Menggunakan Convolutional Neural Networks

Muhammad Zaydan Athallah - 13521104<sup>1</sup>

Program Studi Teknik Informatika

Sekolah Teknik Elektro dan Informatika

Institut Teknologi Bandung, Jl. Ganessa 10 Bandung 40132, Indonesia

<sup>1</sup>13521104@std.stei.itb.ac.id

**Abstract**—Gender, sebagai aspek sosial yang kompleks, sering kali termanifestasi dalam ciri-ciri visual wajah. Dalam era digital ini, pemrosesan gambar telah menjadi salah satu bidang yang berkembang pesat, terutama dalam konteks kecerdasan buatan. Salah satu aspek yang menarik adalah klasifikasi gender berbasis gambar menggunakan Convolutional Neural Network (CNN) dengan landasan teori graf. Makalah ini bertujuan untuk mengidentifikasi dan mengklasifikasikan jenis kelamin seseorang berdasarkan gambar wajah dengan menggunakan pendekatan deep learning. Metode CNN dipilih karena kemampuannya dalam mengekstraksi fitur yang kompleks dari gambar, sehingga dapat memberikan representasi yang baik untuk klasifikasi gender.

**Keywords**—Graf, Convolutional Neural Network, Deep Learning, Gender.

## I. PENDAHULUAN

"Gender" mengacu pada peran sosial, perilaku, dan atribut yang masyarakat anggap sesuai dengan laki-laki atau perempuan. Gender tidak hanya mencakup aspek biologis (seperti jenis kelamin), tetapi juga melibatkan konsep sosial dan budaya yang membedakan antara maskulinitas dan feminitas.

Dalam konteks ini, "jenis kelamin" mengacu pada karakteristik biologis seseorang, seperti kromosom, organ reproduksi, dan hormon. Jenis kelamin dapat dibagi menjadi laki-laki dan perempuan.

Pemodelan teori graf pada model klasifikasi gender menggunakan Convolutional Neural Networks (CNN) adalah sebuah pendekatan inovatif yang menggabungkan konsep matematika diskrit dengan kecanggihan teknologi dalam bidang pembelajaran mesin. Gender, sebagai aspek sosial dan identitas, menjadi fokus penelitian yang semakin mendalam, dan penerapan model ini bertujuan untuk mengembangkan metode yang lebih canggih dalam mengklasifikasikan gender berdasarkan fitur visual terutama pada wajah.

Teori graf memberikan dasar struktural yang kuat untuk merepresentasikan hubungan antara berbagai entitas atau elemen dalam suatu dataset. Dengan menggabungkan teori graf dalam model klasifikasi gender, kita dapat menggambarkan dan menganalisis kompleksitas hubungan antara fitur-fitur visual yang dapat merentang dari wajah hingga elemen lain

dalam gambar.

Convolutional Neural Networks, yang telah terbukti berhasil dalam tugas penglihatan komputer, menjadi bagian integral dari pendekatan ini. CNN dapat secara otomatis mengekstraksi fitur-fitur penting dari data visual, memungkinkan model untuk belajar pola-pola kompleks yang berkaitan dengan atribut gender.

## II. LANDASAN TEORI

### A. Graf

Graf adalah struktur data yang terdiri dari simpul-simpul (atau node) yang terhubung oleh tepi (atau edge). Graf digunakan untuk merepresentasikan hubungan antara objek-objek tersebut.

Graf  $G$  didefinisikan oleh dua himpunan, yakni himpunan  $V$  yang berisi simpul-simpul (node) yang tidak kosong, dan himpunan  $E$  yang berisi sisi-sisi (edge).

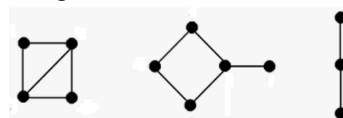
Tepi dapat memiliki arah (graf berarah) atau tidak memiliki arah (graf tidak berarah). Jika tepi memiliki arah graf disebut graf berarah dan jika tidak memiliki arah disebut graf tidak berarah.

Graf dapat diperluas dengan konsep bobot atau label pada tepi, yang memberikan informasi tambahan tentang hubungan antara simpul-simpul. Terdapat beberapa jenis graf yang umum, seperti graf berarah, graf tidak berarah, graf berbobot, dan lainnya.

Graf dapat diklasifikasikan menjadi dua jenis berdasarkan keberadaan sisi ganda atau gelang, yaitu graf sederhana dan graf tak-sederhana.

- Graf sederhana (*simple graph*).

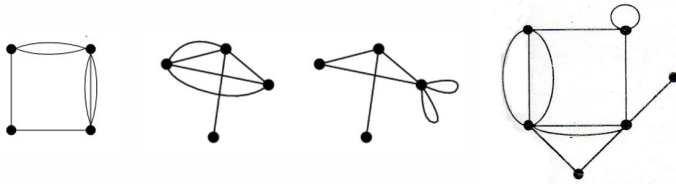
Graf yang tidak memiliki sisi ganda atau gelang disebut graf sederhana.



Gambar 1 Graf Sederhana

Sumber : [1]

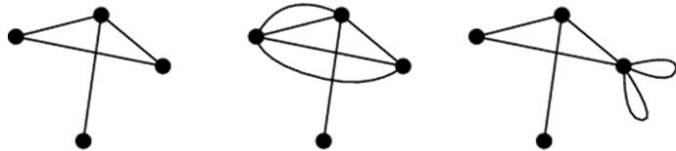
- Graf tak-sederhana (*unsimple graph*)  
Graf yang memiliki sisi ganda atau gelang disebut sebagai graf tak sederhana.



**Gambar 2** Graf tak-sederhana  
Sumber : [1]

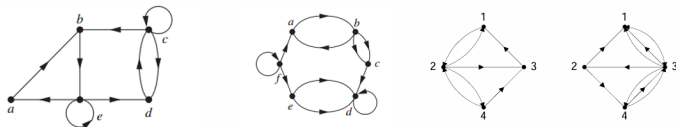
Berdasarkan orientasi arah pada sisi atau tepi, graf dibedakan menjadi dua jenis, yaitu graf berarah dan graf tidak berarah.

- Graf tak-berarah (*undirected graph*)  
Graf yang memiliki sisi atau tepi tanpa orientasi arah disebut graf tidak berarah.



**Gambar 3** Graf tak-berarah  
Sumber : [1]

- Graf berarah (*directed graph*)  
Graf yang setiap sisi atau tepinya memiliki orientasi arah disebut sebagai graf berarah.



**Gambar 4** Graf berarah  
Sumber : [1]

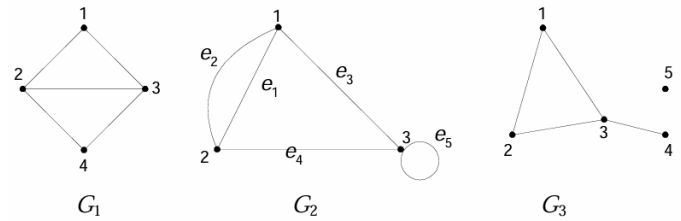
Graf memiliki sejumlah istilah dasar yang berkaitan dengan konsepnya. Dalam konteks ini, berikut adalah beberapa istilah dasar yang terkait dengan graf.

### 1. Ketetangaan (*Adjacent*)

Simpul disebut bertetangga jika terdapat tepi atau sisi yang langsung menghubungkan keduanya dalam suatu graf. Konsep ini menunjukkan bahwa hubungan tetangga antara dua simpul dapat terjadi melalui koneksi langsung oleh tepi atau sisi dalam struktur graf tersebut. Keberadaan tepi ini mengindikasikan adanya keterhubungan langsung antara simpul-simpul yang bersangkutan, mencirikan relasi tetangga dalam konteks graf. Selain itu, sifat bertetangga ini penting dalam menganalisis struktur jaringan dan keterkaitan antara elemen-elemen dalam graf.

Analisis graf  $G_1$  :

- simpul 1 bertetangga dengan simpul 2 dan 3.
- simpul 1 tidak bertetangga dengan simpul 4.



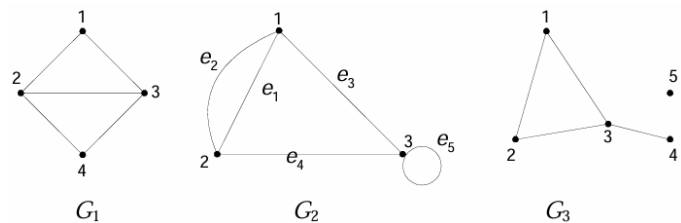
**Gambar 5** Ketetangaan pada Graf  
Sumber : [1]

### 2. Bersisian (*Incidency*)

Irisan dalam graf didefinisikan sebagai situasi di mana tepi  $e=(u,v)$  menunjukkan bahwa tepi tersebut bersentuhan dengan simpul  $u$  dan simpul  $v$ .

Analisis graf  $G_1$ :

- sisi (2, 3) bersisian dengan simpul 2 dan simpul 3.
- sisi (2, 4) bersisian dengan simpul 2 dan simpul 4.
- sisi (1, 2) tidak bersisian dengan simpul 4.



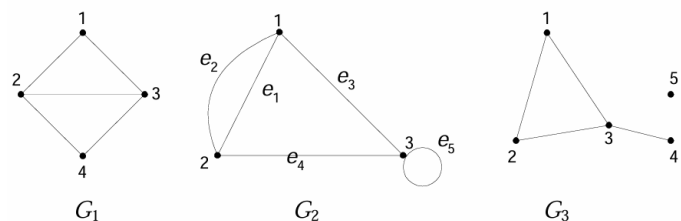
**Gambar 6** Irisan pada Graf  
Sumber : [1]

### 3. Simpul Terpencil (*Isolated Vertex*)

Simpul terpencil adalah simpul yang tidak memiliki tepi yang terhubung langsung dengannya.

Analisis graf  $G_3$ :

- simpul 5 adalah simpul terpencil.



**Gambar 7** Simpul Terpencil  
Sumber : [1]

### 4. Graf Kosong (*null graph or empty graph*)

Graf yang semua sisi tepinya kosong disebut sebagai graf nol atau graf hampa.



**Gambar 8** Graf Kosong  
Sumber : [1]

5. Derajat (*Degree*)

Derajat suatu simpul dalam graf didefinisikan sebagai jumlah tepi yang terhubung langsung dengan simpul tersebut.

Analisis graf G1:

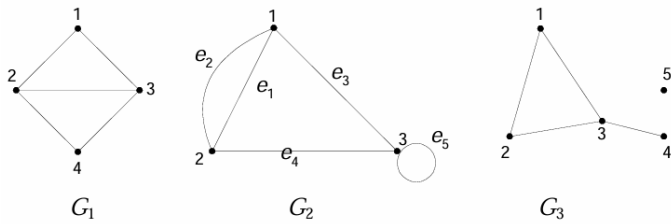
- $d(1) = d(4) = 2$
- $d(2) = d(3) = 3$

Analisis graf G3:

- $d(5) = 0 \rightarrow$  simpul terpencil
- $d(4) = 1 \rightarrow$  simpul anting-anting (pendant vertex)

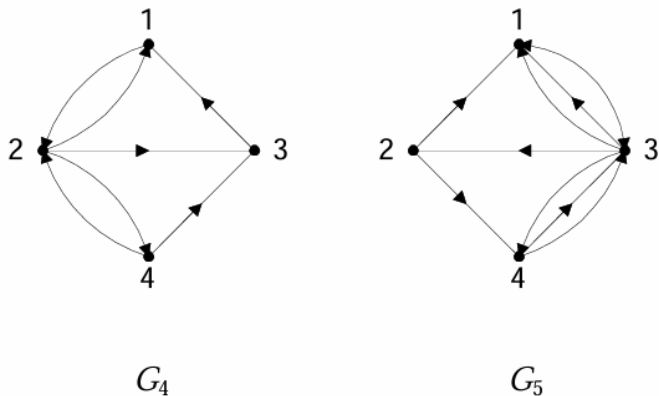
Analisis graf G2:

- $d(1) = 3 \rightarrow$  bersisian dengan sisi ganda
- $d(3) = 4 \rightarrow$  bersisian dengan sisi gelang (*loop*)



**Gambar 9** Derajat Graf  
Sumber : [1]

Dalam konteks graf berarah, konsep derajat simpul terbagi menjadi dua, yaitu derajat masuk (in-degree) dan derajat keluar (out-degree). Derajat masuk mencerminkan jumlah tepi yang mengarah ke suatu simpul, sementara derajat keluar menggambarkan jumlah tepi yang keluar dari simpul tersebut. Dengan kata lain, dalam graf berarah, derajat masuk dan derajat keluar memberikan informasi mengenai seberapa banyak tepi yang terhubung ke atau keluar dari suatu simpul.



**Gambar 10** Derajat Graf Berarah  
Sumber : [1]

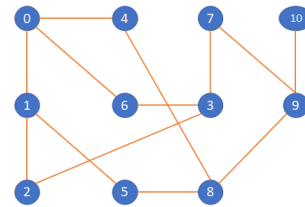
Analisis graf G4:

- $din(1) = 2; dout(1) = 1$
- $din(2) = 2; dout(2) = 3$
- $din(3) = 2; dout(3) = 1$
- $din(4) = 1; dout(3) = 2$

6. Lintasan (*Path*)

Sebuah lintasan yang memiliki panjang  $n$  dari simpul awal  $v_0$  ke simpul tujuan  $v_n$  dalam graf  $G$  adalah urutan simpul-simpul dan sisi-sisi yang bergantian, seperti  $v_0, e_1, v_1, e_2, v_2, \dots, v_{n-1}, e_n, v_n$ , dengan syarat bahwa setiap sisi  $e_1 = (v_0, v_1), e_2 = (v_1, v_2), \dots, e_n = (v_{n-1}, v_n)$  merupakan sisi-sisi dari graf  $G$ .

Panjang lintasan adalah jumlah sisi dalam lintasan tersebut. Lintasan 0, 6, 3, 7, 9, 10 pada  $G$  memiliki panjang 5.



**Gambar 11** Lintasan Graf  
Sumber : [1]

7. Siklus (*Cycle*) atau Sirkuit (*Circuit*)

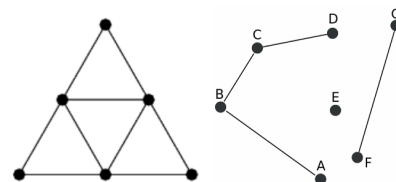
Sirkuit atau siklus dalam graf adalah lintasan yang dimulai dan berakhir pada simpul yang sama.

Analisis Gambar 11 :

Panjang sirkuit adalah jumlah sisi dalam sirkuit tersebut. Sirkuit 0, 4, 8, 5, 1, 0 pada  $G$  memiliki panjang 5.

8. Keterhubungan (*Connected*)

Simpul  $v_1$  dan  $v_2$  dianggap saling terhubung dalam suatu graf jika terdapat lintasan yang menghubungkan  $v_1$  ke  $v_2$ . Sebuah graf dikategorikan sebagai graf terhubung apabila setiap pasangan simpul  $v_i$  dan  $v_j$  dalam himpunan  $V$  memiliki lintasan yang menghubungkan  $v_i$  ke  $v_j$ . Jika kondisi ini tidak terpenuhi, maka graf  $G$  akan disebut sebagai graf tak-terhubung atau *disconnected graph*.



**Gambar 12** Graf Terhubung dan Tidak Terhubung  
Sumber : [1]

Graf berarah  $G$  dianggap terhubung jika graf

tak-berarah yang dapat dihasilkannya juga terhubung. Untuk mendapatkan graf tak-berarah dari  $G$ , langkah yang dapat diambil adalah menghapus arah pada setiap tepinya.

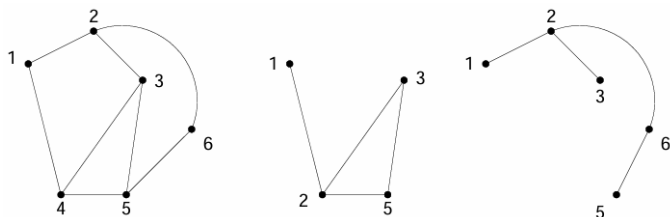
Simpul  $u$  dan  $v$  dalam graf berarah  $G$  dianggap terhubung kuat apabila terdapat jalur satu arah dari  $u$  ke  $v$  dan sekaligus terdapat jalur satu arah dari  $v$  ke  $u$ .

Jika simpul  $u$  dan  $v$  tidak memiliki koneksi kuat satu sama lain dalam graf berarah, tetapi terhubung dalam graf tidak berarah yang berasal dari graf tersebut, maka simpul  $u$  dan  $v$  disebut terhubung lemah (*weakly connected*).

Graf berarah  $G$  diklasifikasikan sebagai graf terhubung kuat (*strongly connected graph*) jika untuk setiap pasangan simpul  $u$  dan  $v$  yang ada dalam  $G$ , terdapat jalur yang menghubungkan keduanya secara kuat. Sebaliknya, jika kondisi tersebut tidak terpenuhi untuk setiap pasangan simpul sembarang dalam  $G$ , maka  $G$  disebut sebagai graf terhubung lemah. Dengan kata lain, dalam graf terhubung lemah, mungkin ada pasangan simpul yang tidak terhubung kuat, meskipun terhubung dalam bentuk graf tidak berarahnya.

9. Upagraf (*Subgraph*) dan Komplemen Upagraf

Anggaplah  $G$  sebagai sebuah graf dengan himpunan simpul  $V$  dan himpunan tepi  $E$ .  $G_1$ , yang dinyatakan sebagai  $(V_1, E_1)$ , adalah upagraf dari  $G$  jika himpunan simpul  $V_1$  adalah subhimpunan dari  $V$  dan himpunan tepi  $E_1$  adalah subhimpunan dari  $E$ . Komplemen dari upagraf  $G_1$  terhadap graf  $G$  didefinisikan sebagai  $G_2 = (V_2, E_2)$ , di mana  $E_2$  adalah selisih (perbedaan) antara himpunan tepi  $G$  dan  $G_1$  ( $E_2 = E - E_1$ ), dan  $V_2$  adalah himpunan simpul yang bersisian dengan himpunan tepi  $E_2$ .



(a) Graf  $G_1$  (b) Sebuah upagraf (c) komplemen dari upagraf

**Gambar 13** Graf, Upagraf, dan Komplemen Upagraf

Sumber : [1]

Dalam suatu graf  $G$ , komponen graf, atau yang disebut juga *connected component*, merujuk pada sejumlah maksimum upagraf yang saling terhubung. Dalam situasi di mana graf tersebut berarah, istilah komponen terhubung kuat (*strongly connected component*) digunakan untuk menyebut

jumlah maksimum upagraf yang memiliki keterhubungan kuat satu sama lain.

10. Upagraf Merentang (*Spanning Subgraph*)

Upagraf  $G_1$  yang disimbolkan sebagai  $(V_1, E_1)$ , yang berasal dari graf  $G$  yang disimbolkan sebagai  $(V, E)$ , dikategorikan sebagai upagraf rentang jika himpunan simpulnya,  $V_1$ , sama dengan himpunan simpul  $G$  (artinya,  $G_1$  mencakup seluruh simpul dari  $G$ ). Dengan kata lain, upagraf rentang mengandung semua simpul yang ada dalam graf aslinya.

11. *Cut-Set*

Cut-set dari graf terhubung  $G$  adalah himpunan tepi yang, ketika dihapus dari  $G$ , menyebabkan  $G$  menjadi tidak terhubung. Oleh karena itu, cut-set selalu menghasilkan dua komponen terpisah.

12. Graf Berbobot (*Weighted Graph*)

Graf berbobot merujuk pada graf di mana setiap tepinya memiliki nilai atau bobot tertentu.

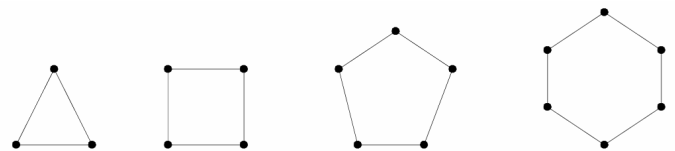
Graf yang menampilkan pola atau struktur tertentu dikenal sebagai graf khusus, dan jenis graf ini sering diterapkan dalam berbagai konteks aplikasi. Berikut beberapa contoh dari graf khusus tersebut.

1. Graf Lengkap (*Complete Graph*)

Graf lengkap adalah jenis graf sederhana di mana setiap simpul terhubung langsung dengan semua simpul lainnya. Graf lengkap dengan  $n$  simpul dilambangkan sebagai  $K_n$ . Jumlah tepi pada graf lengkap yang memiliki  $n$  simpul adalah  $n(n-1)/2$ .

2. Graf Lingkaran

Graf lingkaran, yang juga dikenal sebagai graf sirkuit atau cycle graph, adalah jenis graf sederhana yang memiliki sifat khusus: setiap simpul dalam graf ini memiliki derajat dua. Graf lingkaran dengan  $n$  simpul sering dilambangkan sebagai  $C_n$ . Dalam graf ini, setiap simpul terhubung dengan dua simpul lainnya, membentuk suatu sirkuit tertutup tanpa cabang. Oleh karena itu, derajat setiap simpul pada graf lingkaran adalah dua. Notasi  $C_n$  menggambarkan bahwa graf ini merupakan lingkaran dengan  $n$  simpul yang membentuk siklus tertutup.



**Gambar 14** Graf Lingkaran

Sumber : [1]

3. Graf Teratur (*Regular Graphs*)

Sebuah graf dianggap teratur jika setiap simpul dalam graf tersebut memiliki derajat yang

sama. Jika derajat setiap simpul adalah  $r$ , maka graf tersebut dikenal sebagai graf teratur derajat  $r$ . Jumlah tepi pada graf teratur dengan derajat  $r$  pada setiap simpul dapat dihitung sebagai setengah dari perkalian antara jumlah simpul ( $n$ ) dan derajat setiap simpul ( $r$ ), yaitu  $nr/2$ .

#### 4. Graf Bipartite (*Bipartite Graph*)

Sebuah graf  $G$  dianggap sebagai graf bipartit jika himpunan simpulnya dapat dipisahkan menjadi dua himpunan bagian, yaitu  $V_1$  dan  $V_2$ , sedemikian rupa sehingga setiap tepi dalam  $G$  menghubungkan simpul dari himpunan  $V_1$  ke simpul di himpunan  $V_2$ . Notasi yang digunakan untuk menyimbolkan graf bipartit adalah  $G(V_1, V_2)$ .

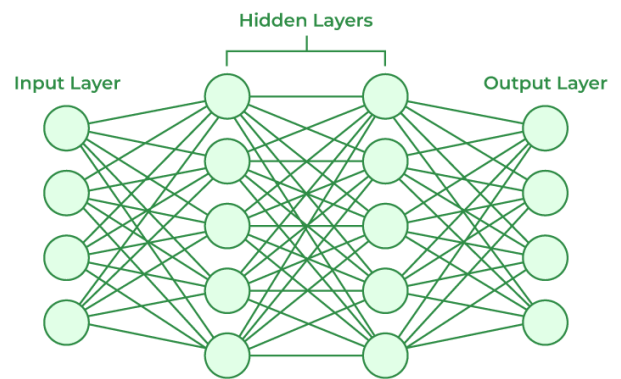
#### B. Artificial Neural Network (ANN)

*Artificial Neural Network* (ANN) atau Jaringan Saraf Tiruan adalah model komputasi terinspirasi oleh struktur dan fungsi jaringan saraf biologis manusia. ANN terdiri dari unit-unit pemrosesan sederhana yang disebut neuron atau node, yang diorganisir dalam lapisan-lapisan. Setiap koneksi antar neuron memiliki bobot yang dapat mempengaruhi satu neuron terhadap neuron lainnya.

Struktur dasar ANN terdiri dari tiga jenis lapisan:

1. **Input Layer (Lapisan Masukan):** Lapisan ini menerima data atau informasi masukan.
2. **Hidden Layers (Lapisan Tersembunyi):** Lapisan-lapisan ini memproses informasi dan melakukan komputasi internal. ANN dapat memiliki satu atau lebih lapisan tersembunyi, tergantung pada kompleksitas masalah yang dihadapi.
3. **Output Layer (Lapisan Keluaran):** Lapisan ini menghasilkan hasil atau prediksi berdasarkan hasil komputasi dari lapisan-lapisan sebelumnya.

ANN dilatih dengan memberikan contoh-contoh data pada masukan dan keluaran yang diinginkan, dan selama proses pelatihan, bobot-bobot antar neuron disesuaikan agar model dapat mempelajari pola atau hubungan dalam data tersebut. ANN digunakan untuk berbagai tugas seperti klasifikasi, regresi, pengenalan pola, dan pemrosesan bahasa alami. ANN merupakan bagian dari bidang studi yang lebih luas yang dikenal sebagai machine learning dan deep learning.

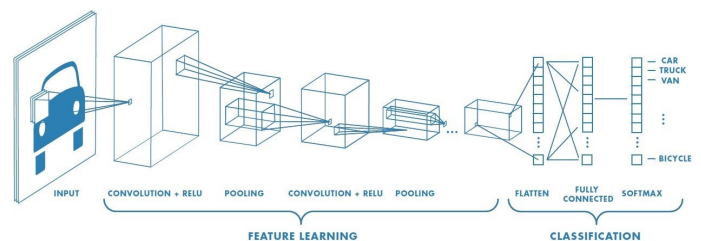


**Gambar 15** Artificial Neural Network

Sumber : [2]

#### C. Convolutional Neural Network (CNN)

*Convolutional Neural Network* (CNN) atau Jaringan Saraf Konvolusional adalah salah satu jenis arsitektur *neural network* yang dirancang khusus untuk memproses dan menganalisis data berbentuk grid, seperti gambar dan video. CNN sangat efektif dalam tugas-tugas pengenalan gambar dan pengolahan visual lainnya. Kelebihan utamanya terletak pada kemampuannya untuk secara otomatis memahami dan mengekstraksi fitur-fitur hierarkis dari data visual.



**Gambar 16** Convolutional Neural Network

Sumber :

Berikut adalah beberapa komponen utama dari CNN:

1. **Layer Konvolusi**  
CNN menggunakan layer konvolusi untuk mengekstraksi fitur dari input gambar. Setiap layer konvolusi memiliki sejumlah filter atau kernel kecil yang bergeser secara konvolutif di seluruh gambar. Filter ini membantu mendeteksi pola dan fitur tertentu seperti tepi, sudut, atau tekstur di berbagai bagian gambar.
2. **Pooling Layer**  
Layer ini digunakan untuk mereduksi dimensi spasial dari output layer konvolusi. Max pooling dan average pooling adalah dua jenis operasi pooling yang umum. Mereka membantu mengurangi ukuran representasi gambar, tetapi tetap mempertahankan informasi penting.
3. **Flatten Layer**  
Layer "Flatten" memiliki peran penting

dalam menghubungkan lapisan konvolusional dengan lapisan *fully connected*, yang biasa disebut sebagai *Dense layer*. Sebelum masuk ke lapisan *fully connected*, hasil keluaran dari lapisan *pooling* diubah menjadi vektor satu dimensi. Proses ini dikenal sebagai "flattening," yaitu mengubah representasi data ke dalam format vektor satu dimensi agar dapat diolah dan digunakan oleh lapisan *fully connected*.

#### 4. Fully Connected Layer (Dense Layer)

Bagian akhir dari CNN yang biasanya terdiri dari satu atau beberapa *fully connected layer*. *Fully connected layer* memungkinkan model memahami pola dan fitur yang kompleks dari data, sambil memperkenalkan non-linearitas melalui fungsi aktivasi. Sebagai bagian akhir dari arsitektur jaringan, dense layer berperan dalam tugas klasifikasi dan regresi dengan menyesuaikan bobot dan bias antar neuron. Dengan fungsi-fungsi ini, dense layer memainkan peran krusial dalam kemampuan jaringan saraf untuk memodelkan dan memahami data secara efektif.

#### 5. Activation

*Activation* adalah fungsi aktivasi yang diterapkan pada output setiap neuron atau node dalam suatu layer. Fungsi aktivasi memperkenalkan unsur non-linearitas ke dalam model, memungkinkan jaringan untuk memahami hubungan yang lebih kompleks dalam data. Tanpa fungsi aktivasi, lapisan-lapisan dalam jaringan saraf hanya akan melakukan transformasi linier pada data input, dan jaringan tersebut tidak akan dapat memodelkan pola yang rumit.

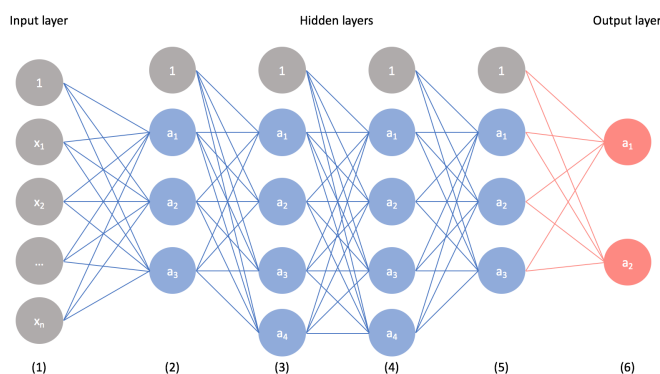
Beberapa contoh fungsi aktivasi yang umum digunakan termasuk:

- ReLU (Rectified Linear Unit): Mengkonversi nilai negatif menjadi nol, tetapi mempertahankan nilai positif.
  - Sigmoid Menghasilkan output di rentang 0 hingga 1, sering digunakan pada lapisan output untuk tugas klasifikasi biner.
  - Tanh Menghasilkan output di rentang -1 hingga 1, sering digunakan untuk menormalisasi data.
  - Softmax Mengkonversi nilai input menjadi distribusi probabilitas, sering digunakan pada lapisan output untuk klasifikasi multi-kelas.
- Fungsi aktivasi memungkinkan jaringan

saraf untuk belajar dan mengekspresikan pola yang lebih kompleks, dan pemilihan fungsi aktivasi bergantung pada karakteristik tugas yang dihadapi oleh jaringan.

### III. TEORI GRAF PADA CONVOLUTIONAL NEURAL NETWORK

Convolutional Neural Network (CNN) pada dasarnya direpresentasikan sebagai struktur graf. CNN memiliki neuron atau simpul yang setara dengan elemen dalam graf, dan sinapsis atau tepi yang mewakili hubungan antar-neuron.



**Gambar 17** Ilustrasi Graf pada CNN  
Sumber :

Setiap layer dalam CNN dapat dikatakan sebagai kumpulan graf bipartit, di mana satu himpunan simpul mewakili fitur atau neuron pada layer tersebut, sementara himpunan simpul yang lain merepresentasikan sumber atau tujuan koneksi antar-neuron. Penerapan konsep graf bipartit dapat dijelaskan untuk setiap layer dalam CNN, di mana simpul mewakili fitur atau neuron, dan koneksi antara simpul-simpul tersebut menggambarkan pola pembobotan atau hubungan spasial. Dengan kata lain, setiap layer terdiri dari graf bipartit yang menggambarkan hubungan antara fitur-fitur yang diekstraksi dan bagaimana koneksi tersebut mempengaruhi pemrosesan informasi dalam jaringan.

Teori Graf pada CNN menjadi landasan konseptual yang memungkinkan analisis lebih mendalam terhadap fungsi dan kinerja jaringan saraf konvolusional dalam konteks pemrosesan data visual. Teori graf juga membantu dalam menjelaskan bagaimana setiap layer dalam CNN terhubung, bagaimana informasi disalurkan melalui jaringan, dan bagaimana fitur-fitur diekstraksi dari data visual. Graf bipartit, khususnya, memungkinkan representasi yang jelas tentang hubungan antara fitur-fitur yang terdapat dalam setiap layer.

### IV. KLASIFIKASI GENDER

#### A. Penambahan Data

Untuk membuat sebuah model, data latih merupakan komponen kunci yang sangat penting. Data latih adalah kumpulan contoh atau sampel yang digunakan oleh model untuk mempelajari pola, hubungan, dan karakteristik dari masalah yang ingin dipecahkan. Proses pelatihan model melibatkan memasukkan data latih ke dalam model,



menyesuaikan parameter internalnya, dan mengoptimalkan performa model tersebut.

Data latih harus mencakup representasi yang cukup dan bervariasi dari kasus-kasus yang mungkin dihadapi oleh model di dunia nyata. Kualitas dan representativitas data latih sangat mempengaruhi kemampuan model untuk menggeneralisasi dan memberikan prediksi yang akurat pada data yang belum pernah dilihat sebelumnya.

Data latih yang digunakan adalah data *open source* yang ada di internet [4]. Data latih ini memiliki 23999 foto dan data tes yang digunakan memiliki 6001 foto.

### B. Eksplorasi Data

Data yang digunakan memiliki 2 kategori yaitu *female* dan *male*. Dalam kasus ini, dengan dua kategori, yaitu "female" dan "male," kita berurusan dengan tugas klasifikasi biner, di mana tujuan utamanya adalah mengelompokkan setiap sampel data ke dalam salah satu dari dua kategori tersebut.

**Tabel 1** Distribusi Data.

Kategori	Jumlah
<i>Female</i>	15154
<i>Male</i>	8845

### C. Data Preprocessing

Sebelum melakukan implementasi model CNN (Convolutional Neural Network), penting untuk mempersiapkan dan memproses data gambar agar sesuai dengan kebutuhan model dan dapat meningkatkan performa model. Tahap preprocessing data gambar terbagi menjadi beberapa tahap, yaitu:

#### 1. Rescale

Setiap gambar akan dilakukan rescale 1./255 yaitu normalisasi nilai piksel pada gambar. Angka 255 merepresentasikan nilai maksimum yang dapat ditemui pada saluran warna (channel) gambar (misalnya, saluran merah, hijau, dan biru pada gambar berwarna). Dengan membagi setiap nilai piksel dalam gambar dengan 255, kita melakukan proses rescaling atau normalisasi sehingga nilai piksel berada dalam rentang antara 0 dan 1.

#### 2. Data Augmentation

Data *augmentation* adalah teknik yang digunakan dalam pemrosesan gambar untuk meningkatkan variasi dataset latih dengan membuat variasi dari data yang sudah ada. Tujuannya adalah untuk membantu model *machine learning* atau *deep learning* untuk memahami dan menggeneralisasi pola dengan lebih baik, serta mengurangi risiko *overfitting*.

Dengan data *augmentation*, kita membuat variasi baru dari setiap gambar dalam dataset latih

dengan menerapkan transformasi kecil atau perubahan pada gambar-gambar tersebut. Transformasi ini dapat mencakup rotasi, pergeseran, pembalikan horizontal atau vertikal, *zoom in* atau *out*, serta perubahan warna atau kecerahan.

### D. Implementasi Model CNN

```
model = tf.keras.models.Sequential([
    # Conv2D digunakan untuk mengekstrak atribut pada gambar data
    tf.keras.layers.Conv2D(32, (3,3), activation='relu', input_shape=(100, 150, 3)),
    # max pooling digunakan untuk mengurangi resolusi gambar data
    tf.keras.layers.MaxPooling2D(2, 2),
    tf.keras.layers.Conv2D(64, (3,3), activation='relu'),
    tf.keras.layers.MaxPooling2D(2, 2),
    tf.keras.layers.Conv2D(128, (3,3), activation='relu'),
    tf.keras.layers.MaxPooling2D(2, 2),
    tf.keras.layers.Conv2D(512, (3,3), activation='relu'),
    tf.keras.layers.MaxPooling2D(2, 2),
    tf.keras.layers.Flatten(),
    tf.keras.layers.Dense(512, activation='relu'),
    tf.keras.layers.BatchNormalization(),
    tf.keras.layers.Dropout(0.5),
    tf.keras.layers.Dense(1, activation='sigmoid')
])
```

**Gambar 18** Lapisan Layer Model CNN  
Sumber : Arsip Pribadi

Model CNN diimplementasikan dengan menggunakan TensorFlow Keras. Model ini dirancang untuk tugas klasifikasi biner, di mana inputnya adalah gambar dengan dimensi 100x150 piksel dan 3 saluran warna (RGB).

Arsitektur model Sequential dimulai dengan lapisan Conv2D pertama yang menggunakan 32 filter dengan ukuran kernel 3x3 dan fungsi aktivasi ReLU. Lapisan ini berfungsi sebagai penyaring untuk mengekstrak atribut atau fitur-fitur visual dari gambar input. Selanjutnya, lapisan MaxPooling2D dengan ukuran pool 2x2 diterapkan untuk mereduksi dimensi gambar dan mengambil fitur-fitur yang paling penting.

Proses ini diulang dengan menambahkan lapisan Conv2D dan MaxPooling2D yang semakin kompleks untuk mengekstrak fitur-fitur yang semakin abstrak. Jumlah filter pada lapisan Conv2D berturut-turut adalah 64, 128, dan 512, dengan pengurangan dimensi menggunakan MaxPooling2D yang sesuai.

Setelah serangkaian lapisan konvolusi dan pooling, dilanjutkan dengan lapisan Flatten untuk mengubah representasi data ke dalam vektor satu dimensi. Selanjutnya, ada lapisan Dense dengan 512 neuron dan fungsi aktivasi ReLU, yang menyediakan kapasitas untuk memproses informasi yang telah diekstrak.

Ditambahkan lapisan BatchNormalization untuk meningkatkan stabilitas dan percepatan pelatihan, diikuti oleh lapisan Dropout dengan tingkat dropout 0.5 untuk mengurangi risiko *overfitting*. Akhirnya, lapisan Dense terakhir dengan satu neuron dan fungsi aktivasi sigmoid digunakan untuk menghasilkan output model sebagai probabilitas untuk klasifikasi biner.

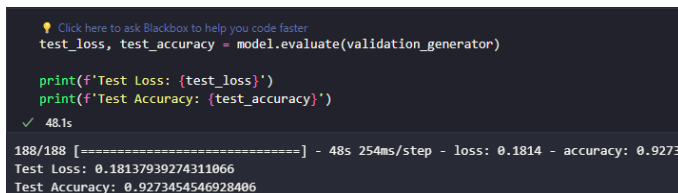
Dengan arsitektur ini, model CNN dapat dilatih untuk memahami dan mengklasifikasikan gambar input sesuai

dengan tugas klasifikasi biner yang ditetapkan.

#### E. Evaluasi

Dalam tahap prediksi, model digunakan untuk melakukan prediksi pada 6001 gambar yang telah disiapkan sebelumnya. Model tersebut telah melalui proses pelatihan dengan jumlah epoch yang lebih tinggi, yakni sebanyak 100 kali.

Model ini berhasil mencapai akurasi sebesar 92.73% dengan validation loss sebesar 18.14% pada tahap evaluasi. Akurasi sebesar 92.73% mengindikasikan bahwa model mampu dengan baik dalam mengklasifikasikan data uji, sedangkan validation loss sebesar 18.14% menunjukkan seberapa baik model dapat melakukan generalisasi ke data yang belum pernah dilihat sebelumnya. Meskipun akurasi tinggi, tetapi nilai validation loss yang cukup tinggi mungkin menandakan adanya overfitting atau masalah lain yang perlu dieksplorasi lebih lanjut.



```
Click here to ask Blackbox to help you code faster
test_loss, test_accuracy = model.evaluate(validation_generator)

print(f'Test Loss: {test_loss}')
print(f'Test Accuracy: {test_accuracy}')
✓ 48.1s
188/188 [=====] - 48s 254ms/step - loss: 0.1814 - accuracy: 0.9273
Test Loss: 0.18137939274311066
Test Accuracy: 0.9273454546928406
```

**Gambar 19** Akurasi Model CNN  
Sumber : Arsip Pribadi

#### V. KESIMPULAN

Kesimpulan yang dapat ditarik dari makalah ini adalah bahwa terdapat banyak penerapan praktis dari konsep Matematika Diskrit di berbagai sektor. Salah satu contohnya adalah pemanfaatan struktur data graf pada Convolutional Neural Network (CNN) untuk tujuan klasifikasi gender secara otomatis. Model ini dapat diintegrasikan dengan suatu aplikasi lain atau dengan alat elektronik untuk menghasilkan klasifikasi secara *real-time*.

#### VI. UCAPAN TERIMA KASIH

Penulis mengungkapkan rasa syukur kepada Tuhan Yang Maha Esa atas kesempatan untuk menyelesaikan makalah ini. Penulis juga ingin menyampaikan terima kasih yang tulus kepada dosen K1, Ibu Nur Ulfa Maulidevi, yang dengan sepenuh hati, berbagi pengetahuannya kepada penulis dan rekan-rekan sekelas di K1.

#### REFERENCES

[1] Munir, Rinaldi. 2016. Matematika Diskrit. Edisi Revisi Keenam. Bandung: Informatika Bandung.

[2] Artificial Neural Networks and its Applications. <https://www.geeksforgeeks.org/artificial-neural-networks-and-its-applications/> diakses 9 Desember 2023 12:00 WIB

[3] A Comprehensive Guide to Convolutional Neural Networks — the ELI5 way. <https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53>

diakses 9 Desember 2023 12:00 WIB

[4] Convolutional neural networks. <https://www.jeremyjordan.me/convolutional-neural-networks/> diakses 9 Desember 2023 12:00 WIB

[5] A Comprehensive Guide to Convolutional Neural Networks. <https://www.v7labs.com/blog/convolutional-neural-networks-guide> diakses 9 Desember 2023 15:00 WIB

#### PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 10 Desember 2023



Muhammad Zaydan Athallah 13521104