

# Klasifikasi Lukisan Karya Van Gogh Menggunakan *Convolutional Neural Network- Support Vector Machine*

<http://dx.doi.org/10.28932/jutisi.v7i1.3399>

Riwayat Artikel

Received: 11 Februari 2021 | Final Revision: 19 Maret 2021 | Accepted: 25 Maret 2021

Yohannes✉<sup>#1</sup>, Daniel Udjulawa<sup>#2</sup>, Febbiola<sup>#3</sup>

<sup>#</sup>Teknik Informatika, STMIK Global Informatika MDP  
Jalan Rajawali No. 14 Palembang

<sup>1</sup>yohannesmasterous@mdp.ac.id

<sup>2</sup>daniel@mdp.ac.id

<sup>3</sup>febbiolafebbi@gmail.com

**Abstract** — Painting is a work of art with various strokes, textures and color gradations so that a painting that is synonymous with beauty is created. The various paintings created have characteristics, such as the paintings by Van Gogh, which have tightly arranged strokes, creating a repetitive and patterned impression. This study classifies paintings by Van Gogh or not by using the VGG-19 and ResNet-50 feature extraction methods. The SVM method is used as a classification method with two optimizations, namely random and grid optimization in the linear kernel. The data set used consisted of 124 Van Gogh paintings and 207 paintings by other painters. The use of VGG-19 feature extraction using grid optimization has the best value of 93,28% using the use of random optimization which has a value of 92,89%. The use of ResNet-50 using grid optimization with the best value of 90,28% using the use of random optimization which has a value of 90,15%. The extraction feature of VGG-19 is better than ResNet-50 in paintings by Van Gogh or not.

**Keywords** — Classification; Painting; VGG-19; ResNet-50; Linear; Random; Grid.

## I. PENDAHULUAN

Lukisan adalah karya seni yang menggunakan kanvas, kertas, kuas, dan cat dengan berbagai goresan, tekstur, dan gradasi warna [1]. Setiap lukisan tersebut identik dan dapat direpresentasikan terutama keindahan dan maknanya [2]. Beragam lukisan yang diciptakan memiliki karakteristik, salah satu pelukis terkenal di dunia *pasca-impresionisme* asal Belanda adalah Van Gogh. Aliran seni lukis yang dilukis oleh Van Gogh yaitu *Post Impressionisme*, *Pointilisme* [3], *Fauvisme*, dan *Ekspressionisme* [4].

Perkembangan teknologi saat ini tidak terlepas dari karya seni yang diciptakan oleh para pelukis, dengan adanya kemajuan dari teknologi saat ini, dapat mengklasifikasikan

objek dari lukisan karya Van Gogh berdasarkan karakteristik yang ada pada objek tersebut. Salah satu objek yang dapat dikenali karakteristiknya adalah lukisan.

Klasifikasi pengenalan lukisan karya Van Gogh dilakukan karena lukisan tersebut memiliki goresan yang tersusun rapat sehingga menciptakan repetitif dan kesan berpola yang dipertegas. Hal ini menjadi tantangan sendiri, terutama tantangan yang ditujukan untuk mengklasifikasi lukisan karya Van Gogh sehingga mendapatkan hasil pengukuran karakteristik pola pada lukisan karya Van Gogh atau bukan dengan penerapan komputasi [5].

Jenis metode yang digunakan untuk melakukan pengenalan klasifikasi adalah SVM (*Support Vector Machine*). SVM merupakan teknik untuk melakukan prediksi, dalam kasus regresi dan kasus klasifikasi, dan juga memiliki suatu prinsip dasar *linear classifier*, yang artinya SVM dapat dipisahkan secara *linear*, SVM juga dikembangkan untuk bekerja pada *problem non-linear* dengan mencari *hyperplane* pada ruang kerja dengan konsep *kernel* [6].

CNN (*Convolutional Neural Network*) merupakan salah satu *neural network* dan dikenal sebagai *deep learning*. Arsitektur yang dirancang memiliki kemampuan *feature engineering* sehingga dapat membangun model ekstraksi fitur dalam peningkatan akurasi [7], seperti menetapkan bobot dan bias ke berbagai objek dalam gambar dengan memanfaatkan proses *convolution* untuk menggerakkan *kernel convolution (filter)* dengan ukuran tertentu ke dalam gambar serta mendapatkan hasil representatif yang baru berupa informasi. Awalnya, CNN dirancang untuk memproses data citra secara efisien dan dikembangkan dengan properti seperti konektivitas lokal, invariansi spasial, dan fitur hierarki. CNN telah mendorong terobosan di beberapa bidang penelitian dan diterapkan seperti

psychiatry dan neurology untuk menyelidiki gangguan otak [8].

Penelitian yang terkait dengan metode CNN-SVM, seperti klasifikasi lalat buah dengan *background* kompleks yang menghasilkan tingkat akurasi sebesar 92,04% [9], Penggunaan metode CNN-SVM sudah dilakukan di bidang lainnya seperti klasifikasi pengenalan angka tulisan tangan mendapatkan akurasi sebesar 94,40% [10], pengenalan bentuk 3D dengan ModelNet10 [11], diagnosa kesalahan bantalan dibawah kebisingan lingkungan dengan akurasi sebesar 99,5% [12], klasifikasi penginderaan jauh dengan tingkat akurasi sebesar 93,5% [13], pengenalan citra MRI dengan tingkat akurasi sebesar 99,5% [14], dan klasifikasi gambar dengan tingkat akurasi 90,72% [15].

Arsitektur model CNN-SVM sendiri dirancang dengan mengganti lapisan keluaran terakhir model CNN-SVM. Model SVM memiliki akurasi klasifikasi serta kemampuan untuk generalisasi yang sangat baik [9]. Model CNN-SVM berbeda dengan CNN, dimana Softmax *classifier* dalam struktur CNN diganti dengan model pengklasifikasi SVM. Metode ini mampu menyelesaikan masalah *overfitting* pada model CNN serta meningkatkan hasil akurasi dalam pengenalan sebesar 92,04% [9]. Cara kerja struktur model *hybrid* CNN-SVM, yaitu inputan *citra* yang telah dinormalisasi kemudian dipusatkan selanjutnya dikirim ke dalam input layer serta CNN asli dengan lapisan keluaran yang telah dilatih dengan proses pelatihan [10].

Penggunaan metode *hybrid* CNN-SVM ini lebih baik dibandingkan CNN saja untuk citra yang memiliki kemiripan dalam kerapatan piksel, karena lukisan bersifat seni dan keseluruhan area dalam objek sangat penting, yang artinya tidak bisa mendapatkan fitur, sehingga akan digunakan metode *neural network* untuk mendapatkan pola lukisan dari setiap pelukis. Penggunaan metode klasifikasi biasa belum bisa dilakukan karena memerlukan fitur untuk memastikan lukisan tersebut merupakan lukisan karya Van Gogh atau bukan. Mengingat lukisan Van Gogh cukup sulit dibandingkan pola pada umumnya yang mudah dilihat secara langsung.

## II. LANDASAN TEORI

### A. Van Gogh

Vincent Willem van Gogh, anak dari seorang pastor dari Dutch Reformed Church yang bernama Theodorus van Gogh (1822 sampai 1885), dan ibunya Van Gogh yang bernama Anne Cornelia, nee Carbentus (1819 sampai 1907), Van Gogh lahir di Groot Zundert, tepatnya di Netherlands pada 30 Maret 1853. Van Gogh pernah membuat lukisan bergaya *Post-Impressionisme*, seperti *View of Paris from Montmartre*, *Paris Seen from Vincent's Room in the Rue Lepic*, dan *Terrace of the Cafè "La Guinguette"*, serta lukisan bergaya *Ekspressionisme*, seperti *The Church in Auvers* [3].

### B. Karya Van Gogh

#### 1. Self Portrait

*Self Portrait* merupakan lukisan potret diri yang dibuat menggunakan minyak di atas kanvas pada September 1889. Lukisan ini merupakan karya terakhir Van Gogh sebelum meninggalkan *Saint Remy de Provence* di Selatan Prancis yang berada di *Musee d'Orsay*, Paris [16]. Lukisan *Self Portrait* dapat dilihat pada Gambar 1.



Gambar 1. Self Portrait [17]

#### 2. Peasant Woman Cooking

*Peasant Woman Cooking* merupakan karya yang dibuat di Nuene pada akhir musim semi 1885. Warna gelap pada lukisan Van Gogh ini mengingatkan pelukis pada kentang yang berdebu tidak dikupas [18]. Lukisan *Peasant Woman Cooking* dapat dilihat pada Gambar 2.



Gambar 2. Peasant Woman Cooking [17]

#### 3. The Irises

*The Irises* adalah lukisan yang dibuat oleh Van Gogh saat di R. S. Jiwa *Saint Paul de Mausole, Saint Remy de Provence*, Prancis menjelang kematiannya tahun 1890. Hal ini menggambarkan bahwa dapat mencegah dari kegilaan dengan tetap meneruskan melukis. Lukisan ini berada di Museum Getty, Los Angeles [19]. Lukisan *The Irises* dapat dilihat pada Gambar 3.



Gambar 3. The Irises [17]

#### 4. The Potato Eaters

*The Potato Eaters* adalah lukisan pemakan kentang yang dilukis pada April 1885 di Nuene, museum Amsterdam. Lukisan ini memperlihatkan kehidupan para petani yang hanya mampu menyantap kentang

diterangi lampu minyak untuk bertahan hidup [20]. Lukisan *The Potato Eaters* dapat dilihat pada Gambar 4.



Gambar 4. The Potato Eaters [17]

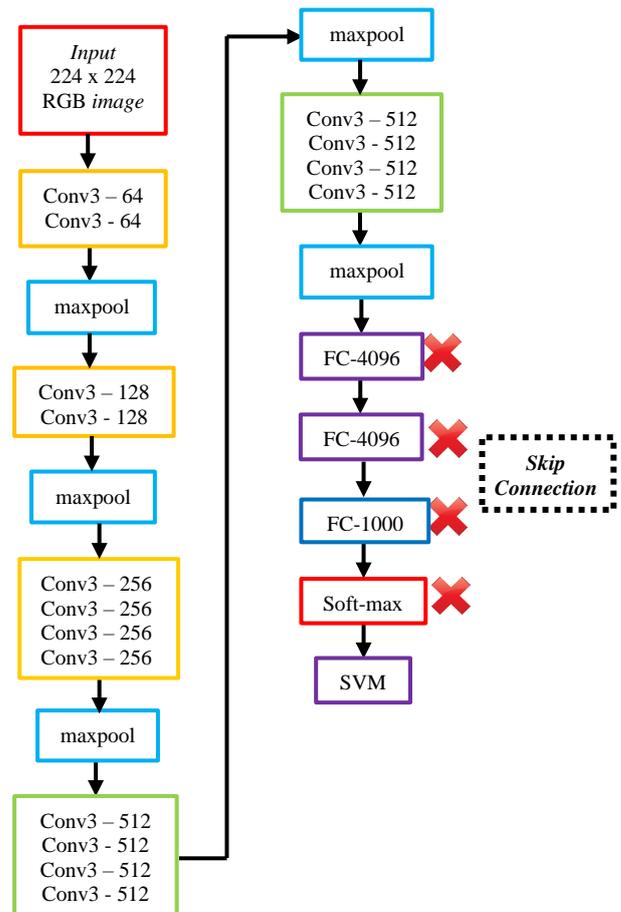
### C. CNN-SVM

CNN (*Convolutional Neural Network*) adalah salah satu kelas *deep feed forward* pada *artificial neural networks* dan sudah banyak diterapkan pada bagian analisis citra, biasanya digunakan untuk data *image* atau mendeteksi dan mengenali objek pada sebuah *image*. CNN terdiri atas *input layer*, *hidden layers*, dan *output layer*. *Hidden layer* terdiri dari *convolution layers*, *pooling layers*, *normalization layers*, *ReLU layer*, *fully connected layers*, dan *loss layer* [21]. *Pooling layer* berfungsi untuk menjaga ukuran data pada saat konvolusi, serta melakukan *downsampling*, sehingga data dikembangkan menjadi bagian yang lebih kecil, mudah dikelola, dan mudah dalam mengatur *overfitting*. *Normalization layer* dibuat untuk mengatasi perbedaan batas nilai pada input citra, karena kemungkinan hasil yang kecil sehingga tidak digunakan. *Loss layer* adalah lapisan terakhir CNN, yang menentukan pelatihan atas penyimpangan antara label dan hasil prediksi. Berbagai macam *loss function*, yaitu *softmax loss* yang berfungsi dalam memprediksi satu dari beberapa kelas, *sigmoid cross-entropy loss* dalam prediksi probabilitas nilai pada interval [0,1], dan *Euclidean loss* yang berfungsi dalam regresi nilai kontinu [7].

Penggunaan arsitektur VGG-19 memiliki pengenalan klasifikasi gambar dalam skala yang besar, dengan melakukan *filter* konvolusi yang sangat kecil berukuran 3x3 sehingga menunjukkan peningkatan yang signifikan. Model VGG-19 ini memiliki 19 *layer* yang telah memberikan hasil yang bagus dalam pengembangan modelnya dan termasuk 5 *top classification error* dalam *validation set* dan *test set* [22].

Model yang sudah ada tersebut digunakan lagi, kemudian dimasukkan lukisan karya Van Gogh atau bukan untuk dipelajari. Selama *training*, memasukkan gambar 224 x 224 RGB dengan ukuran yang tetap. Proses awalnya dengan mengurangi nilai RGB rata-rata dengan perhitungan pada pelatihan di setiap piksel. Gambar tersebut melewati tumpukan *convolution layer* dan menggunakan *convolution filter* yang berukuran 3x3. Proses yang telah dilakukan tersebut melakukan *skip connection*, yaitu melewati proses di posisi FC-4096, FC-1000, dan *Soft-max*, karena hanya memerlukan proses sebelumnya untuk diambil ekstraksi fitur dan diberikan kepada SVM untuk dilakukan pelatihan, sehingga saat prosesnya telah selesai dilakukan menggunakan kernel *linear* yang telah dimodifikasi tersebut dan dapat mengklasifikasi lukisan karya Van Gogh atau

bukan. Skema *Hybrid CNN-SVM* pada arsitektur VGG-19 dapat dilihat pada Gambar 5.

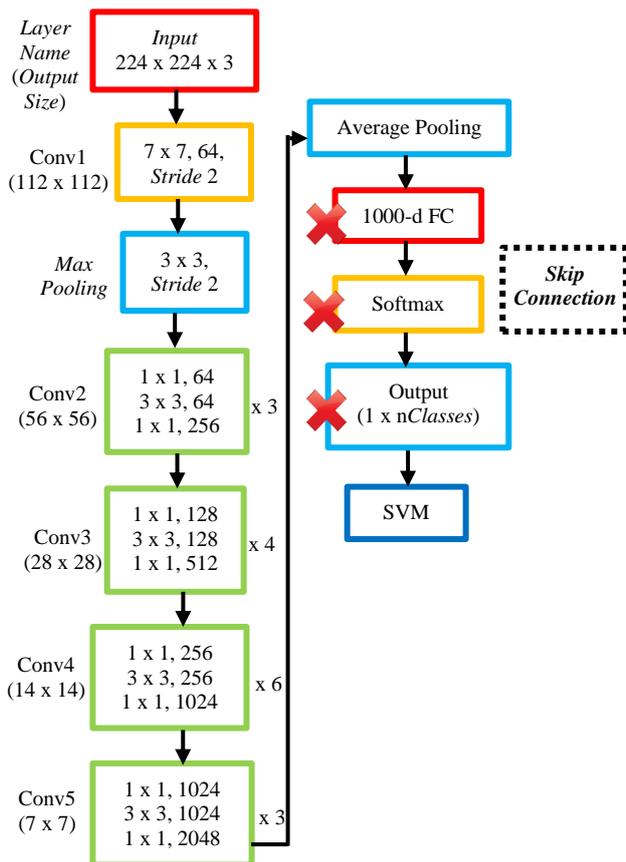


Gambar 5. Skema *Hybrid CNN-SVM* Pada VGG-19.

Pada Gambar 6, penggunaan arsitektur ResNet-50 dalam CNN digunakan karena arsitektur ResNet-50 menggunakan konsep *shortcut connections* yang dapat menggunakan kembali fitur sebagai masukan dilapisan sebelumnya terhadap lapisan keluaran, sehingga dapat mengurangi hilangnya fitur penting saat melakukan konvolusi. Penggunaan ResNet-50 sendiri lebih baik dibandingkan penggunaan lainnya, seperti teknik *processing* segmentasi karena lukisan yang dilihat itu secara keseluruhan, sehingga proses segmentasi tidak digunakan karena saat berbicara tentang lukisan semua *area* yang ada di dalam kanvas adalah objek goresan yang penting dalam seniman sehingga tidak bisa dipotong.

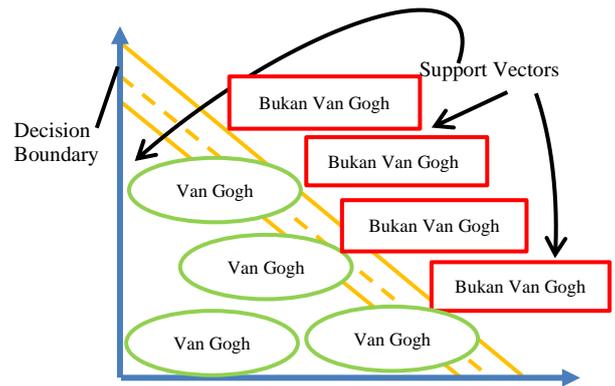
Pada arsitektur ResNet-50 dimasukkan gambar berukuran 224 x 224 *pixels* kemudian dilakukan proses *convolution* dengan lima tahapan, kemudian dilanjutkan dengan *average pooling* untuk mengambil nilai rata-rata dengan mengurangi dimensi yang terdiri dari *filter* dan ukuran *stride* yang telah ditentukan sebelumnya. *Stride* di skema *hybrid CNN-SVM* pada ResNet-50 ini memiliki nilai 2, yang artinya melakukan pergeseran matrix sebanyak dua dimensi. Proses yang akan

dilakukan selanjutnya, yaitu melakukan *skip connection*, yaitu melewati proses di posisi 1000-d FC, *Soft-max*, dan *Output* ( $1 \times nClasses$ ) karena hanya memerlukan proses sebelumnya untuk diambil ekstraksi fitur nya dan diberikan kepada SVM untuk dilakukan pelatihan, sehingga saat prosesnya telah selesai dilakukan menggunakan kernel *linear* yang telah dimodifikasi tersebut dan dapat mengklasifikasi lukisan karya Van Gogh atau bukan menggunakan arsitektur ResNet-50. Skema *Hybrid CNN-SVM* pada arsitektur ResNet-50 dapat dilihat pada Gambar 6.



Gambar 6. Skema *Hybrid CNN-SVM* Pada ResNet-50.

SVM merupakan metode dalam *supervised learning* untuk klasifikasi (*Support Vector Classification*) dan regresi (*Support Vector Regression*). SVM mengatasi masalah klasifikasi dan regresi dengan *linear* dan *non-linear*. SVM digunakan untuk klasifikasi data dua kelas dan diperluas untuk *Multi Class SVM* [7]. SVM terdiri dari beberapa jenis diantaranya *Binary Class SVM* (klasifikasi dua kelas), *Multi Class SVM* (klasifikasi lebih dari dua kelas), dan *Fuzzy SVM* (klasifikasi yang lebih baik dalam menangani lebih dari dua kelas) [7]. Cara kerja SVM dapat dilihat pada Gambar 7.



Gambar 7. Cara Kerja SVM

SVM digunakan untuk mencari optimum untuk memaksimalkan jarak antar kelas. *Hyperplane* adalah sebuah fungsi yang digunakan untuk pemisah antar kelas atau sebuah pemisah data, *hyperplane* bisa mengarah pada titik, garis lurus, bidang datar dua dimensi, maupun bidang lainnya yang berdimensi tinggi [7]. SVM yang digunakan dalam penelitian ini adalah *Binary Class SVM*, yaitu dengan menggunakan fungsi *kernel linier* [9], [10].

*Binary Class SVM*, pada *soft margin SVM* menemukan *hyperplane* maksimum dimana linier dipisahkan dengan memecahkan masalah utama dalam pengoptimalan. Misalkan set pelatihan  $(x_i, y_i)$  untuk  $i = 1, 2, 3, \dots$ , sebagai indeks,  $l$  dimana  $x_i \in \mathbb{R}^n$  dan  $y_i \in \{+1, -1\}$ .  $l$  untuk jumlah poin data dalam pelatihan dan  $n$  untuk jumlah fitur nya [23] sehingga perhitungan SVM dapat ditentukan dengan persamaan (1).

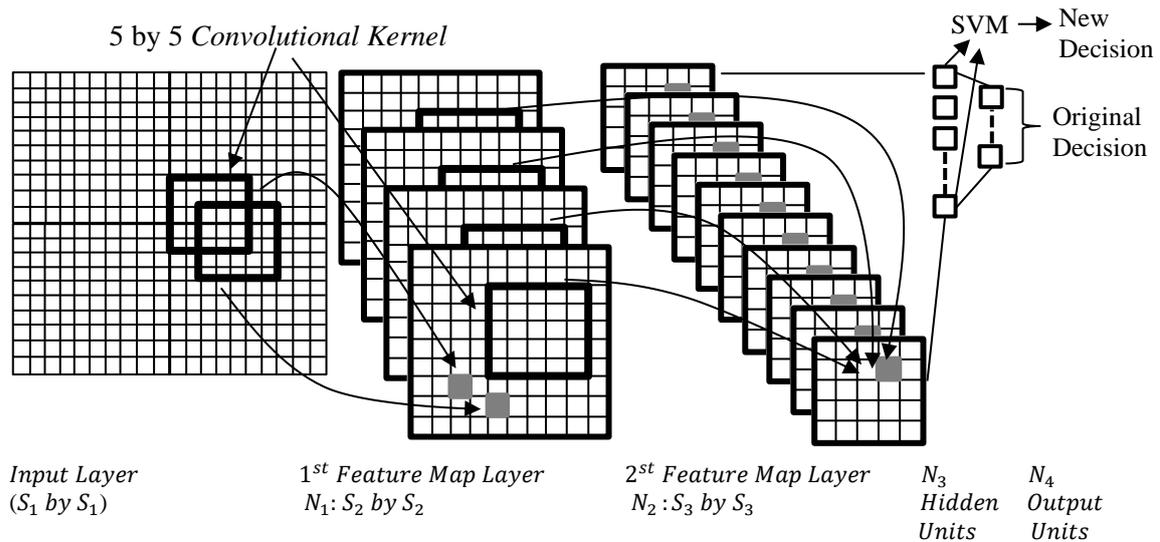
$$\min: \tau(\vec{w}, \vec{\xi}) = \frac{1}{2} ||w^2|| + C \sum_{i=1}^l \xi_i \quad (1)$$

dengan batasan sesuai dengan persamaan (2).

$$y_i w^T x_i \geq 1 - \xi_i, \xi_i \geq 0, \forall i. \quad (2)$$

Dimana  $\xi_i$  adalah *soft margin hyperplane* atau variabel *slack* yang digunakan untuk kemungkinan kesalahan dalam klasifikasi, dengan  $i$  sebagai indeks nya. Parameter  $C$  berfungsi parameter kesalahan [23]. Semakin besar parameter  $C$  nya, maka semakin besar kesalahan dalam klasifikasi, sehingga hanya sedikit data yang bisa melewati batas *hyperplane* tersebut. SVM yang menghitung *dot product* untuk data dalam ruang fitur [7], yang disebut *kernel trick* [10]. SVM dengan *kernel linear* menggantikan file lapisan keluaran. SVM mengambil keluaran dari lapisan tersembunyi sebagai vektor fitur baru untuk pelatihan.

SVM menggunakan dua pendekatan dalam proses klasifikasi data, yaitu meminimalisasi *error* pada data *training* serta meminimalisasi resiko *error*, menggunakan *hyperplane* dengan *margin* terbesar pada data *training* dalam melakukan klasifikasi data [24].



Gambar 8. Cara Kerja CNN-SVM [10]

Arsitektur model *hybrid* CNN-SVM ini dirancang dengan mengganti lapisan keluaran terakhir model CNN-SVM sebagai model klasifikasi. Model SVM memiliki akurasi klasifikasi dan kemampuan generalisasi yang baik. Jadi, model CNN-SVM diusulkan, dimana pengklasifikasi *Softmax* diganti dengan SVM *classifier* dalam struktur CNN. Ini memecahkan masalah *overfitting* model CNN dan meningkatkan akurasi pengenalan [9].

Unit keluaran dari lapisan terakhir di jaringan CNN, dengan perkiraan probabilitas untuk sampel masukan. Setiap probabilitas keluaran dihitung dengan fungsi aktivasi. Masukan dari fungsi aktivasi adalah kombinasi *linear* dari keluaran ke *hidden layer* sebelumnya dengan bobot yang dapat dilatih, ditambah dengan bias.

Cara kerja struktur model *hybrid* CNN-SVM, yaitu gambar *input* yang sudah dinormalisasi dan dipusatkan kemudian dikirim ke *input layer*, CNN yang asli dengan *output layer* dilatih dengan proses pelatihan. SVM dengan *kernel linear* kemudian menggantikan *file* lapisan keluaran serta mengambil keluaran didalam lapisan tersembunyi untuk dijadikan sebagai vektor fitur baru pelatihan. Setelah pengklasifikasi SVM memiliki data yang terlatih, maka akan dibuat keputusan baru untuk menguji gambar melalui ekstrak fitur. Sehingga model *hybrid* CNN-SVM baru ini akan mengungguli setiap klasifikasi yang didasarkan pada fakta bahwa *hybrid* baru dibuat dan dilatih [10]. Untuk lapisan terakhir CNN yang terhubung tersebut sepenuhnya diganti dengan SVM dalam pelatihan untuk mengklasifikasi lukisan karya Van Gogh atau bukan. Menggunakan 331 data latih untuk nilai dari lapisan jaringan. CNN yang sudah terlatih kemudian digunakan sebagai vektor fitur baru dalam mewakili setiap pola masukan kepada SVM untuk pembelajaran dan pengujian. Membangun model SVM secara *hybrid*, digunakannya *kernel linear* untuk menentukan *kernel* yang paling optimal. Dengan metode validasi pada

*dataset* pelatihan, dengan pemilihan model sebagai proses untuk menemukan file parameter yang optimal.

Penggunaan arsitektur ResNet-50 untuk mengklasifikasi lukisan karya Van Gogh atau bukan. ResNet-50 adalah model *deep residual learning framework*, sehingga setiap *layer network* memiliki referensi kedalam *layer network* sebelumnya, sehingga menjadikan proses optimasi menjadi lebih mudah [25].

Cara kerja CNN-SVM dapat dilihat pada Gambar 8, yaitu *input layer* pada CNN adalah lapisan masukan yang terdiri dari *image* pada *kernel* konvolusi (*filter*) berukuran 5x5 berfungsi untuk menerima data dari luar, kemudian diproses dengan *pooling layer* untuk mengurangi dimensi atau melakukan *downsampling*. *Feature map layer* adalah *output* dari proses konvolusi yang melakukan proses perkalian antara *input* dengan *filter* dan dilakukan *feature map layer* untuk kedua kalinya. Sehingga saat semakin dalam *layer* pada gambar lukisan, maka fitur yang dihasilkan semakin detail pada hasil klasifikasi. *Full connection* adalah *layer* yang melakukan transformasi sehingga dapat mengklasifikasi data ke dalam SVM. *Hidden units* adalah lapisan tersembunyi antara *input layer* dan *output layer* yang mempunyai *input* bobot yang menghasilkan *output unit* [26]. Hasil dari *hidden units* akan membuat *new decision*, CNN terlatih digunakan sebagai vektor fitur baru untuk mewakili setiap pola masukan dan diberikan kepada SVM untuk pembelajaran dan pengujian.

#### D. Metode Perhitungan

##### 1. FAR (False Acceptance Rate)

Perhitungan FAR digunakan untuk melihat sistem melakukan kesalahan dalam penerimaan masukan citra, dimana citra lukisan yang seharusnya ditolak itu diterima dan dimasukkan oleh sistem tersebut [27]. Secara matematis, perhitungan FAR dapat dihitung dengan persamaan (4).

$$FAR = \frac{\text{jumlah data citra yang salah terima}}{\text{jumlah data citra pencocokan}} \times 100\% \quad (4)$$

## 2. Mode

Perhitungan Mode digunakan dengan mengambil nilai yang paling sering muncul dalam data citra atau yang memiliki frekuensi data terbesar dalam perhitungannya. Secara sistematis, perhitungan Mode dapat dihitung dengan persamaan (5).

$$Mo = t_b + \left( \frac{d_1}{d_1 + d_2} \right) c \quad (5)$$

## 3. Sum

Perhitungan Sum digunakan dengan cara menjumlahkan seluruh data citra yang digunakan. Secara sistematis, perhitungan Sum dapat dilakukan pada persamaan (6).

$$Sum = n_1 + n_2 + n_3 + \dots + n_n \quad (6)$$

## 4. Mean

Perhitungan Mean digunakan untuk mencari rata-rata dalam mengklasifikasi lukisan, seperti membagi jumlah data citra dengan banyaknya data citra secara keseluruhan. Secara sistematis, perhitungan Mean dapat dihitung dengan persamaan (7).

$$\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i \quad (7)$$

Dimana  $x_i$  adalah nilai tengah kelas ke  $i$ .

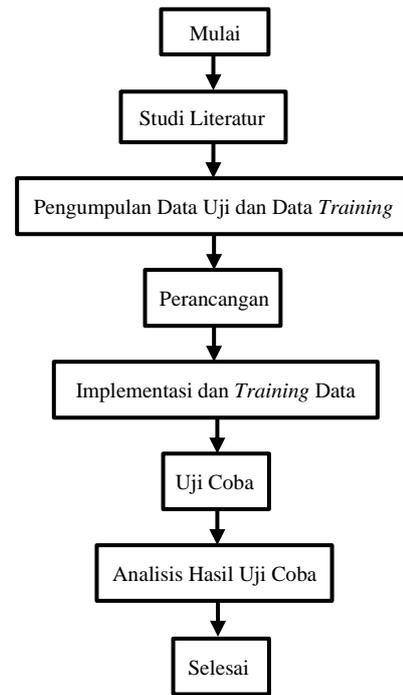
## 5. Median

Perhitungan Median digunakan dengan membagi jumlah data citra menjadi dua bagian atau sama dengan mengambil nilai tengahnya. Secara sistematis, perhitungan Median dapat dihitung dengan persamaan (8).

$$Me = \frac{x_{n+1}}{2} \quad (8)$$

### III. METODOLOGI PENELITIAN

Berikut ini merupakan beberapa tahapan yang perlu dilakukan untuk mengklasifikasi lukisan karya Van Gogh menggunakan CNN-SVM yang dapat dilihat pada Gambar 9.



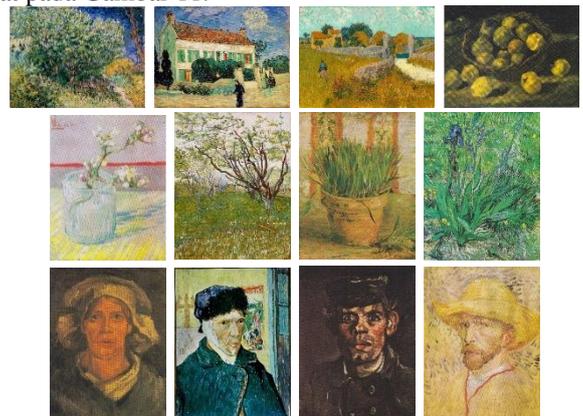
Gambar 9. Metodologi Penelitian

#### A. Studi Literatur

Pada tahapan ini, dilakukan dengan pembelajaran literatur berupa jurnal dan buku yang berkaitan dengan topik penelitian ini, yaitu penggunaan metode CNN-SVM untuk klasifikasi citra lukisan dengan menggunakan faktor berupa gambar, bentuk, ruang, tekstur, dan warna dari objek tersebut.

#### B. Pengumpulan Data Uji dan Data Training

Pada tahap ini dilakukan pengumpulan data *training*, *dataset* yang digunakan dalam penelitian ini berjumlah 124 untuk lukisan karya Van Gogh dan 207 data lukisan dari 94 pelukis lain. Data lukisan karya Van Gogh dapat dilihat pada Gambar 10 dan data lukisan di luar karya Van Gogh dapat dilihat pada Gambar 11.



Gambar 10. Lukisan Karya Van Gogh [17]



Gambar 11. Lukisan Di Luar Karya Van Gogh [17]

### 1. Pembagian Data dan Data Testing

Pada tahapan ini, dilakukan pembagian data *training* dan data *testing*. *Dataset* yang digunakan berjumlah 331 data yang terdiri dari 124 data lukisan dari karya Van Gogh dan 207 data lukisan dari 94 pelukis lain, semua *dataset* dari lukisan karya Van Gogh atau bukan dilakukan teknik *patch image* terlebih dahulu kemudian dilanjutkan *resizing* citra dengan ukuran  $224 \times 224$  pixels [28], untuk setiap citra lukisan yang telah *patch*. Proses ini dilakukan dengan menjalankan *script* pada Python, semua gambar yang terdapat dalam setiap folder dibaca dari *dataset* lukisan kemudian dibuat *directory* baru untuk menyimpan hasilnya. Semua *image* yang telah di *patch* pada data lukisan karya Van Gogh atau bukan menghasilkan 47.408 citra untuk *training* dan 12.537 citra untuk *testing*, dengan total keseluruhan 59.948 citra dari setiap folder *dataset* lukisan, dengan menggunakan *5-fold cross validation*.

### 2. Model Extraction Feature

Pada tahapan ini, penelitian menggunakan dua model *extraction feature*, yaitu VGG-19 dan ResNet-50 sebagai perbandingan dengan penggunaan optimasi *random* dan *grid* pada *kernel linear* didalamnya.

Pada arsitektur ResNet-50 ini sudah digunakan model terlatih dengan *framework caffe* dan dilakukan *train* dengan *dataset ImageNet* serta *normalisasi batch*. Model ResNet-50 yang telah dilatih menggunakan *dataset ImageNet* ini digunakan untuk melakukan ekstraksi fitur pada *dataset* citra lukisan berupa gambar, bentuk, ruang, tekstur, dan warna dari objek tersebut [28].

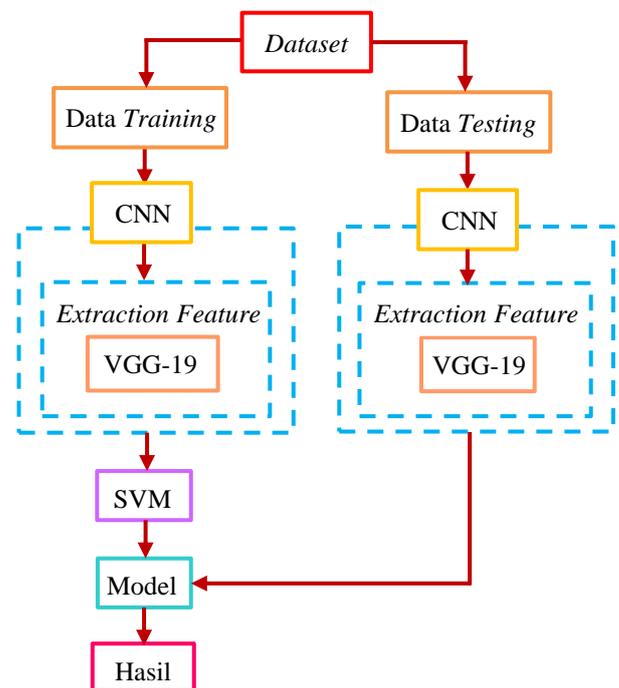
Penggunaan arsitektur ResNet-50 dan VGG-19, keberhasilan *training* penting didalam *normalisasi batch* sehingga dapat mengurangi peningkatan kesalahan yang ada. Tahapan *resizing* yang telah dilakukan pada pembagian data dan data *testing* sebelumnya kemudian digunakan untuk *extraction feature* dari setiap potongan gambar yang telah di *patch*.

Arsitektur VGG-19 memiliki pengenalan gambar dalam skala yang besar, sehingga dilakukan peningkatan dengan *filter* konvolusi yang sangat kecil berukuran  $3 \times 3$  untuk menunjukkan bahwa peningkatan signifikan ini dapat mendorong kedalaman 19 lapisan [22]. Selama *training*,

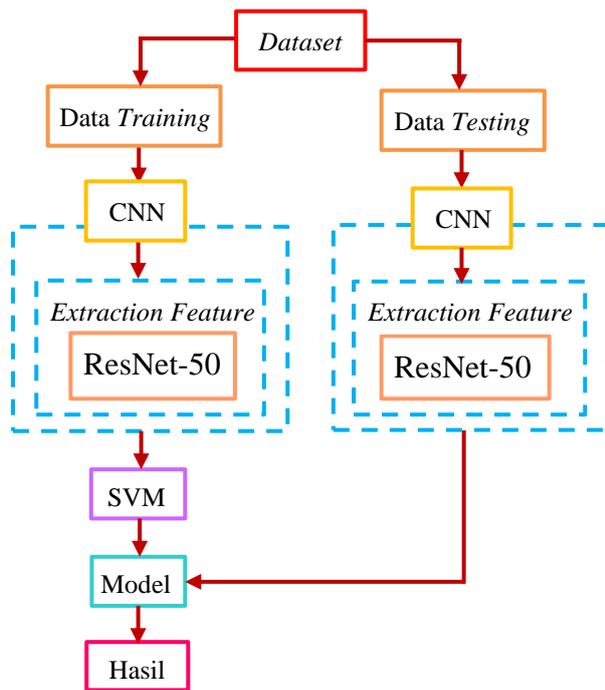
dimasukkan gambar  $224 \times 224$  RGB dengan ukuran yang tetap. Proses awalnya dengan mengurangi nilai RGB rata-rata dengan perhitungan pada pelatihan di setiap piksel. Gambar tersebut melewati tumpukan *convolution layer* dan menggunakan *convolution filter* yang berukuran  $3 \times 3$  dan juga berukuran  $1 \times 1$ .

### C. Perancangan

Pada tahapan ini, dilakukan perancangan penelitian dan sistem yang dibutuhkan, dimulai dengan membagi *dataset* untuk data *training* dan data *testing*, selanjutnya menggunakan teknik *patch image*, artinya membagi setiap gambar lukisan tersebut menjadi beberapa bagian yang lebih kecil [17] dan ukuran setiap potongan gambar lukisan yang dibagi menjadi  $224 \times 224$  pixels. Gambar lukisan yang telah dibagi menjadi beberapa bagian tersebut, kemudian dilakukan ekstraksi fitur menggunakan CNN dengan dua arsitektur yang berbeda, yaitu arsitektur VGG-19 dan ResNet-50 serta penggunaan kernel *linear* dengan dua optimasi, yaitu optimasi *random* dan *grid*. Pada penerapan klasifikasi untuk *patch image*, data *training* membuat model pengklasifikasi *patch* tersebut kemudian data *testing* melakukan perhitungan untuk setiap *patch* dengan beberapa metode perhitungan untuk optimasinya, seperti FAR (*False Acceptance Rate*), Mode, Sum, Mean, dan Median. Skema perancangan klasifikasi lukisan pada arsitektur VGG-19 dan ResNet-50 dapat dilihat pada Gambar 12 dan Gambar 13.



Gambar 12. Skema Perancangan Klasifikasi Lukisan Pada Arsitektur VGG-19



Gambar 13. Skema Perancangan Klasifikasi Lukisan Pada Arsitektur ResNet-50

#### D. Implementasi dan Training Data

Pada tahapan ini, dilakukan implementasi dari penelitian dan sistem yang telah dirancang sebelumnya dengan data *training* yang sebelumnya telah ada agar sistem dapat mengenali dan melakukan klasifikasi terhadap data uji. Tahapan ini juga dilakukan *training* pada sistem agar dapat melakukan klasifikasi pada lukisan, citra diambil untuk dilakukan klasifikasi menggunakan metode CNN-SVM.

#### E. Uji Coba

Pada tahapan ini, dilakukan uji coba terhadap data uji yang telah ada sebelumnya dengan menggunakan sistem yang telah diimplementasi. Tahapan ini dilakukan pengujian dari sistem dalam melakukan klasifikasi lukisan dan hasil dari pengujian tersebut akurat atau tidak akan dicatat untuk dilakukan analisa setelah semua pengujian selesai.

#### F. Analisis Hasil Uji Coba

Pada tahapan ini, dilakukan analisis terhadap hasil uji coba yang telah dilakukan sebelumnya untuk menjawab tujuan dari penelitian ini. Hasil uji coba tersebut dihitung untuk mendapatkan tingkat keberhasilan dari metode yang digunakan dengan *confusion matrix* untuk menghitung nilai *accuracy*, *precision*, dan *recall*, *f1-score* yang dapat dilihat pada persamaan (9), (10), (11), dan (12).

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (9)$$

$$precision = \frac{TP}{TP + FP} \quad (10)$$

$$recall = \frac{TP}{TP + FN} \quad (11)$$

$$f1-score = 2 \times \frac{precision \times recall}{precision + recall} \quad (12)$$

Dari analisis yang dilakukan ini, dapat disimpulkan bahwa seberapa akuratnya penggunaan klasifikasi lukisan karya Van Gogh atau bukan menggunakan CNN-SVM pada arsitektur VGG-19 dan ResNet-50 sebagai *extraction feature* serta pengklasifikasi SVM pada *kernel linear* dalam penggunaan optimasi *random* dan *grid*.

### IV. PENGUJIAN DAN PEMBAHASAN

Hasil pengujian yang dilakukan ini menggunakan dua ekstraksi fitur, yaitu VGG-19 dan ResNet-50 pada *kernel linear* dengan menggunakan dua optimasi, yaitu optimasi *random* dan *grid*, serta menggunakan metode perhitungan untuk optimasinya seperti FAR, Mode, Sum, Mean, dan Median di dalam SVM.

#### A. Implementasi Ekstraksi Fitur VGG-19 Pada Kernel Linear Menggunakan Optimasi Random.

TABEL I  
CONFUSION MATRIX VGG-19 PADA KERNEL LINEAR OPTIMASI RANDOM

Confusion Matrix		Bukan Van Gogh	Van Gogh
FAR	Bukan Van Gogh	39	1
	Van Gogh	3	24
MODE	Bukan Van Gogh	39	2
	Van Gogh	3	23
SUM	Bukan Van Gogh	38	1
	Van Gogh	4	24
MEAN	Bukan Van Gogh	38	1
	Van Gogh	4	24
MEDIAN	Bukan Van Gogh	38	1
	Van Gogh	4	24

Dari Tabel I, dapat dilihat bahwa pengenalan lukisan karya Van Gogh dari TP (*True Positive*), yaitu jumlah data positif yang terklasifikasi dengan benar oleh sistem pada metode perhitungan Mode di *confusion matrix* ini memiliki hasil sebesar 23 yang sedikit lebih kecil dibandingkan dengan metode perhitungan FAR, Sum, Mean, dan Median dengan hasil sebesar 24. Metode perhitungan Mode dari FN (*False Negative*), yaitu jumlah data negatif namun terklasifikasi salah oleh sistem sebesar 2 dan FP (*False Positive*), yaitu jumlah data positif namun terklasifikasi salah oleh sistem sebesar 3. Metode perhitungan Sum, Mean, dan Median sama-sama memiliki nilai TP (*True Positive*), yaitu jumlah data positif yang terklasifikasi dengan benar oleh sistem sebesar 24, dengan nilai FN (*False Negative*), yaitu jumlah data negatif namun terklasifikasi salah oleh sistem sebesar 1 dan untuk nilai FP (*False Positive*), yaitu jumlah data positif namun terklasifikasi salah oleh sistem sebesar 4.

B. Implementasi Ekstraksi Fitur VGG-19 Pada Kernel Linear Menggunakan Optimasi Grid.

TABEL II  
CONFUSION MATRIX VGG-19 PADA KERNEL LINEAR OPTIMASI GRID

Confusion Matrix		Bukan Van Gogh	Van Gogh
FAR	Bukan Van Gogh	39	1
	Van Gogh	3	24
MODE	Bukan Van Gogh	39	3
	Van Gogh	3	22
SUM	Bukan Van Gogh	38	1
	Van Gogh	4	24
MEAN	Bukan Van Gogh	38	1
	Van Gogh	4	24
MEDIAN	Bukan Van Gogh	38	1
	Van Gogh	4	24

Dari Tabel II, dapat dilihat bahwa pengenalan lukisan karya Van Gogh dari TP (*True Positive*), yaitu jumlah data positif yang terklasifikasi dengan benar oleh sistem pada metode perhitungan Mode di *confusion matrix* ini memiliki hasil sebesar 22 yang sedikit lebih kecil dibandingkan dengan metode perhitungan FAR, Sum, Mean, dan Median dengan hasil sebesar 24. Metode perhitungan Mode dari FN (*False Negative*), yaitu jumlah data negatif namun terklasifikasi salah oleh sistem sebesar 3 dan FP (*False Positive*), yaitu jumlah data positif namun terklasifikasi salah oleh sistem sebesar 3. Metode perhitungan Sum, Mean, dan Median sama-sama memiliki nilai TP (*True Positive*), yaitu jumlah data positif yang terklasifikasi dengan benar oleh sistem sebesar 24, dengan nilai FN (*False Negative*), yaitu jumlah data negatif namun terklasifikasi salah oleh sistem sebesar 1 dan untuk nilai FP (*False Positive*), yaitu jumlah data positif namun terklasifikasi salah oleh sistem sebesar 4.

C. Implementasi Ekstraksi Fitur ResNet-50 Pada Kernel Linear Menggunakan Optimasi Random.

TABEL III  
CONFUSION MATRIX RESNET-50 PADA KERNEL LINEAR OPTIMASI RANDOM

Confusion Matrix		Bukan Van Gogh	Van Gogh
FAR	Bukan Van Gogh	37	2
	Van Gogh	5	23
MODE	Bukan Van Gogh	37	2
	Van Gogh	5	23
SUM	Bukan Van Gogh	37	2
	Van Gogh	5	23
MEAN	Bukan Van Gogh	38	2
	Van Gogh	4	23
MEDIAN	Bukan Van Gogh	37	2
	Van Gogh	5	23

Dari Tabel III, dapat dilihat bahwa pengenalan lukisan karya Van Gogh dari TP (*True Positive*), yaitu jumlah data positif yang terklasifikasi dengan benar oleh sistem pada metode perhitungan FAR, Mode, Sum, Mean, dan Median di

*confusion matrix* ini sama-sama memiliki hasil sebesar 23 serta FN (*False Negative*), yaitu jumlah data negatif namun terklasifikasi salah oleh sistem sebesar 2, sedangkan FP (*False Positive*), yaitu jumlah data positif namun terklasifikasi salah oleh sistem sebesar 4 untuk metode perhitungan Mean dan pada metode perhitungan FAR, Mode, Sum, dan Median memiliki hasil sebesar 5.

D. Implementasi Ekstraksi Fitur ResNet-50 Pada Kernel Linear Menggunakan Optimasi Grid.

TABEL IV  
CONFUSION MATRIX RESNET-50 PADA KERNEL LINEAR OPTIMASI GRID

Confusion Matrix		Bukan Van Gogh	Van Gogh
FAR	Bukan Van Gogh	37	2
	Van Gogh	5	23
MODE	Bukan Van Gogh	37	2
	Van Gogh	5	23
SUM	Bukan Van Gogh	37	2
	Van Gogh	5	23
MEAN	Bukan Van Gogh	38	1
	Van Gogh	4	24
MEDIAN	Bukan Van Gogh	37	2
	Van Gogh	5	23

Dari Tabel IV, dapat dilihat bahwa pengenalan lukisan karya Van Gogh dari TP (*True Positive*), yaitu jumlah data positif yang terklasifikasi dengan benar oleh sistem pada metode perhitungan FAR, Mode, Sum, dan Median di *confusion matrix* ini memiliki hasil sebesar 23 yang sedikit lebih kecil dibandingkan metode perhitungan Mean yang memiliki hasil sebesar 24. Metode perhitungan Mean dari FN (*False Negative*), yaitu jumlah data negatif namun terklasifikasi salah oleh sistem sebesar 1, sedangkan FP (*False Positive*), yaitu jumlah data positif namun terklasifikasi salah oleh sistem sebesar 4. Metode perhitungan FAR, Mode, Sum, dan Median memiliki FN (*False Negative*), yaitu jumlah data negatif namun terklasifikasi salah oleh sistem sebesar 1 dan FP (*False Positive*), yaitu jumlah data positif namun terklasifikasi salah oleh sistem sebesar 5.

E. Analisis Hasil Pengujian

Pada Gambar 14, dapat dilihat bahwa perbandingan hasil VGG-19 dengan optimasi *random* ini memiliki nilai *accuracy* sebesar 93,00%, *precision* sebesar 88,00%, dan *recall* sebesar 92,00% yang sedikit lebih kecil dengan menggunakan metode perhitungan Mode, jika dibandingkan dengan metode perhitungan FAR, Sum, Mean, dan Median memiliki nilai yang sama pada *accuracy* sebesar 94,00%. Hasil perhitungan VGG-19 menggunakan *kernel linear* dengan optimasi *random* yang didapatkan secara keseluruhan adalah 92,89%, serta tidak mengalami *overfitting*.

Pada Gambar 15, dapat dilihat bahwa perbandingan hasil VGG-19 dengan optimasi *grid* ini memiliki nilai *accuracy* sebesar 91,00%, *precision* sebesar 88,00%, dan *recall*

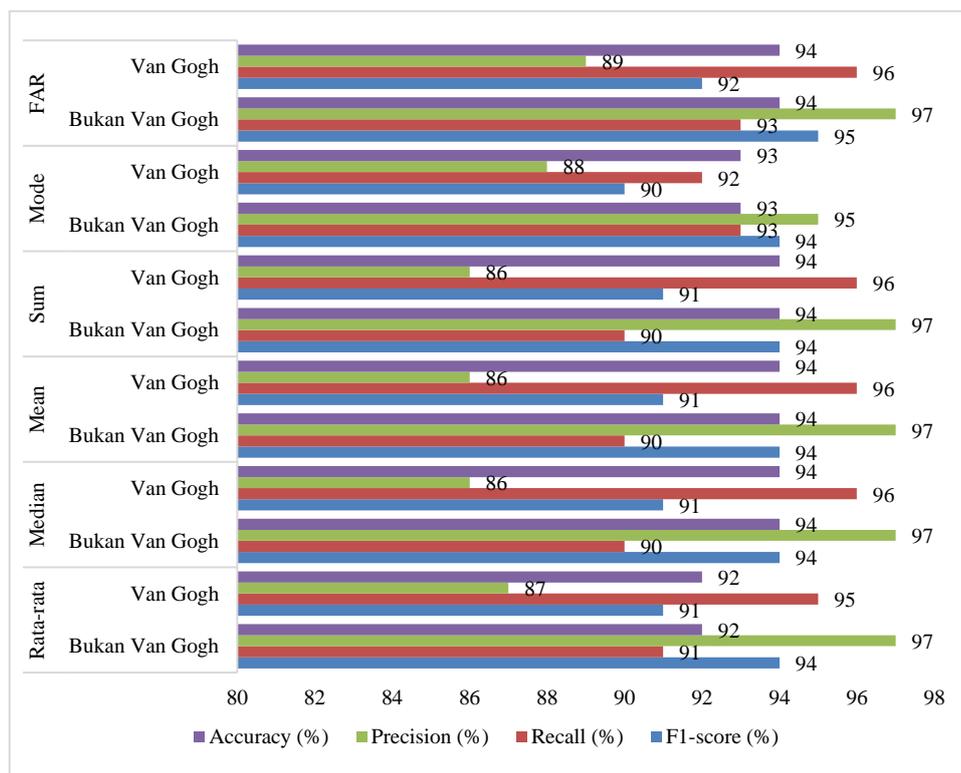
sebesar 88,00% yang sedikit lebih kecil dengan menggunakan metode perhitungan Mode, jika dibandingkan dengan metode perhitungan Sum, Mean, dan Median memiliki nilai yang sama pada *accuracy* sebesar 93,00%. Metode perhitungan FAR memiliki hasil yang lebih besar dengan *accuracy* sebesar 94,00%, *precision* sebesar 89,00%, dan *recall* sebesar 96,00%. Hasil perhitungan VGG-19 menggunakan *kernel linear* dengan optimasi *grid* yang didapatkan secara keseluruhan adalah 93,00%, serta tidak mengalami *overfitting*.

Pada Gambar 16, dapat dilihat bahwa perbandingan hasil ResNet-50 dengan optimasi *random* ini memiliki nilai *accuracy* sebesar 91,00%, *precision* sebesar 85,00%, dan *recall* sebesar 92,00% dengan metode perhitungan Mean, jika dibandingkan dengan metode perhitungan FAR, Mode, Sum, dan Median yang sama-sama memiliki nilai *accuracy* sebesar 90,00% yang sedikit lebih kecil. Hasil *precision* dan *recall* yang didapatkan pada metode perhitungan FAR, Mode, Sum, dan Median memiliki nilai yang sama, yaitu 82,00% untuk *precision* dan 92,00% untuk *recall*. Hasil perhitungan ResNet-50 menggunakan *kernel linear* dengan optimasi *random* yang didapatkan secara keseluruhan adalah 90,15%, serta tidak mengalami *overfitting*.

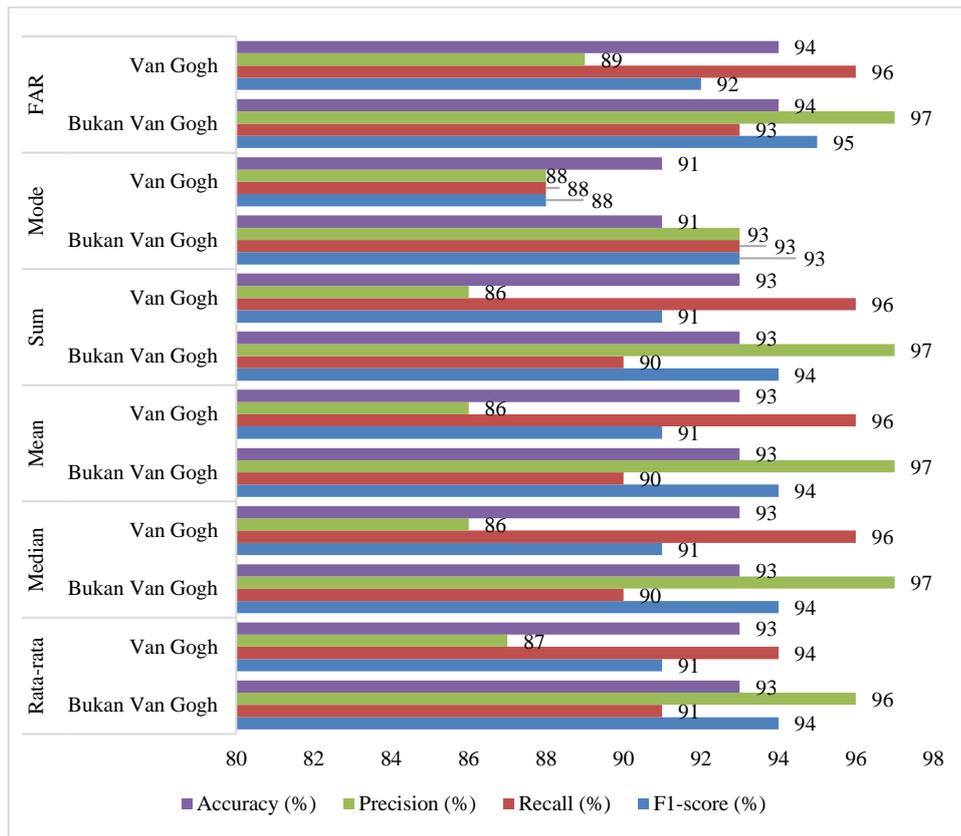
Pada Gambar 17, dapat dilihat bahwa perbandingan hasil ResNet-50 dengan optimasi *grid* ini memiliki nilai *accuracy*

sebesar 93,00%, *precision* sebesar 86,00%, dan *recall* sebesar 96,00% dengan metode perhitungan Mean, jika dibandingkan dengan metode perhitungan FAR, Mode, Sum, dan Median yang sama-sama memiliki nilai *accuracy* sebesar 90,00% yang sedikit lebih kecil. Hasil perhitungan ResNet-50 menggunakan *kernel linear* dengan optimasi *grid* yang didapatkan secara keseluruhan adalah 90,28%, serta tidak mengalami *overfitting*.

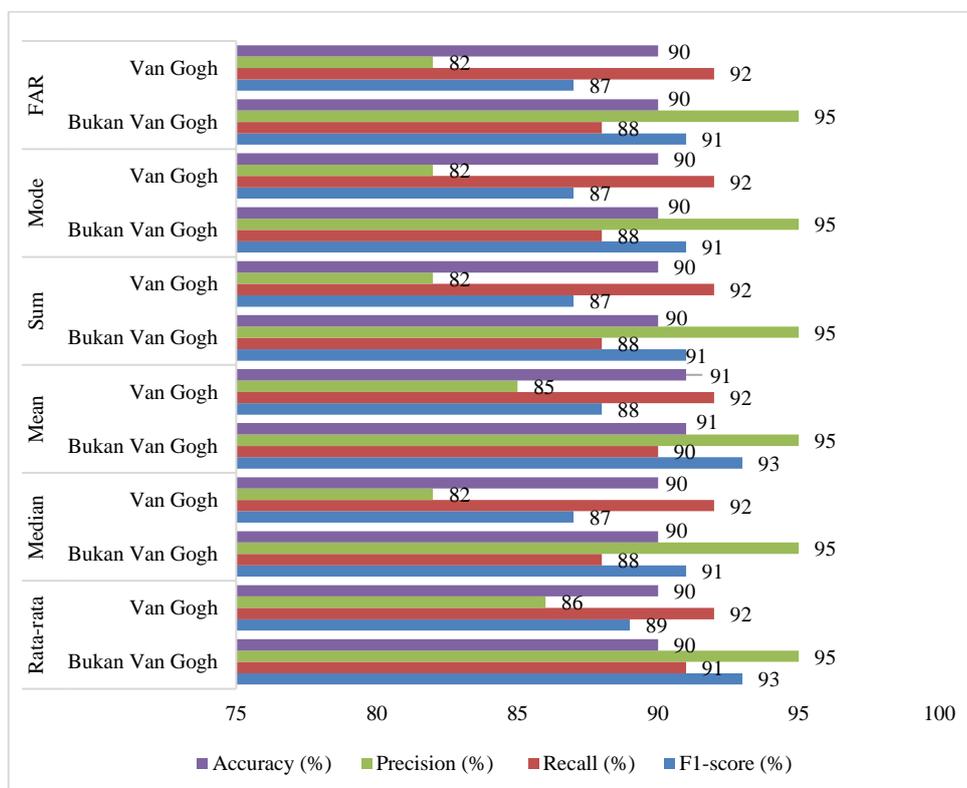
Pada Gambar 18, dapat dilihat bahwa perbandingan hasil fitur pada penggunaan arsitektur VGG-19 dengan optimasi *grid* memiliki hasil *accuracy* sebesar 93,00%, *precision* sebesar 87,00%, dan *recall* sebesar 94,00% lebih baik dibandingkan ResNet-50 dengan optimasi *random* yang memiliki nilai *accuracy* sebesar 90,00%, *precision* sebesar 86,00%, dan *recall* sebesar 92,00%, serta penggunaan optimasi *grid* yang memiliki nilai *accuracy* sebesar 90,00%, *precision* sebesar 83,00%, dan *recall* sebesar 93,00%. Penggunaan arsitektur VGG-19 dengan optimasi *random* memiliki hasil yang sedikit lebih kecil jika dibandingkan dengan penggunaan arsitektur VGG-19 pada optimasi *grid* memiliki hasil *accuracy* sebesar 92,00%, *precision* sebesar 87,00%, dan *recall* sebesar 95,00%.



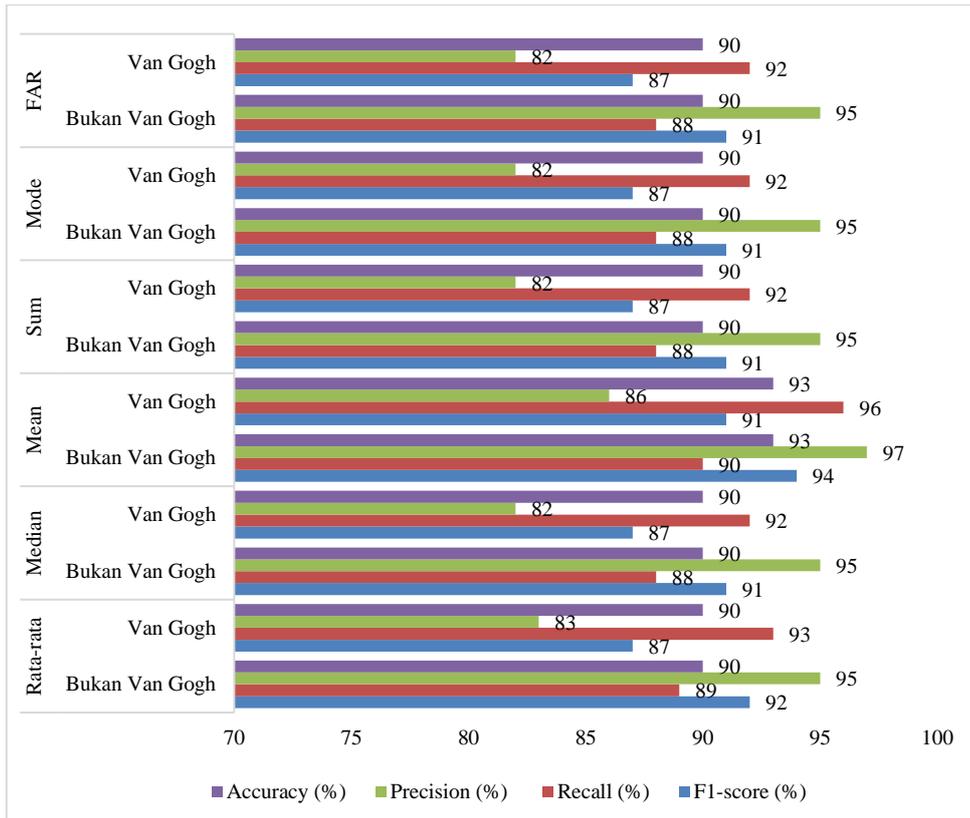
Gambar 14. Perbandingan Hasil VGG-19 Kernel Linear Optimasi Random



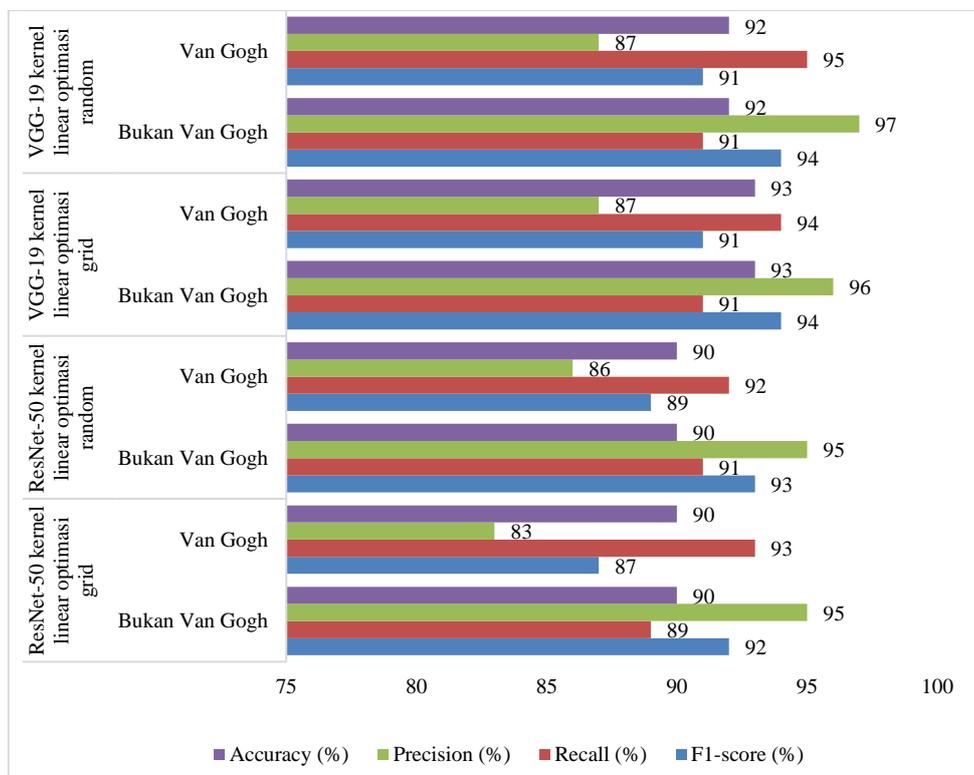
Gambar 15. Perbandingan Hasil VGG-19 Kernel Linear Optimasi Grid



Gambar 16. Perbandingan Hasil ResNet-50 Kernel Linear Optimasi Random



Gambar 17. Perbandingan Hasil ResNet-50 Kernel Linear Optimasi Grid



Gambar 18. Perbandingan Hasil Antar Fitur

## V. SIMPULAN

Mengklasifikasi lukisan karya Van Gogh atau bukan dengan menggunakan teknik *patch image*, artinya membagi setiap gambar lukisan tersebut menjadi beberapa bagian yang lebih kecil [17] dan dilakukan *resizing* citra dengan ukuran 224 x 224 *pixels* pada setiap potongan gambar lukisan. Gambar lukisan yang telah dibagi menjadi beberapa bagian tersebut, kemudian dilakukan ekstraksi fitur menggunakan CNN dengan dua arsitektur yang berbeda, yaitu arsitektur VGG-19 dan ResNet-50 serta penggunaan kernel *linear* dengan dua optimasi, yaitu optimasi *random* dan *grid*. Pada penerapan klasifikasi untuk *patch image*, data *training* membuat model pengklasifikasi *patch* tersebut kemudian data *testing* melakukan perhitungan untuk setiap *patch* dengan beberapa metode perhitungan untuk optimasinya, seperti FAR (*False Acceptance Rate*), Mode, Sum, Mean, dan Median.

Penggunaan ekstraksi fitur VGG-19 dengan *kernel linear* pada optimasi *grid*, memiliki hasil yang terbaik dengan tingkat *accuracy* sebesar 93,00%, *precision* sebesar 87,00%, dan *recall* sebesar 94,00%. Berdasarkan Perbandingan hasil pengujian yang dilakukan, penggunaan ekstraksi fitur VGG-19 dengan menggunakan optimasi *grid* memiliki nilai yang terbaik sebesar 93,28% daripada penggunaan ResNet-50 dengan optimasi *random* yang memiliki hasil sebesar 90,15% maupun penggunaan ResNet-50 dengan optimasi *grid* yang memiliki hasil sebesar 90,28%. Penggunaan VGG-19 pada optimasi *random* memiliki hasil yang sedikit lebih kecil daripada VGG-19 dengan optimasi *grid*, yaitu dengan hasil nilai sebesar 92,89% karena VGG-19 termasuk dalam termasuk 5 *top classification error* dalam *validation set* dan *test set* serta ResNet-50 menggunakan konsep *shortcut connections* yang dapat menggunakan kembali fitur sebagai masukan dilapisan sebelumnya terhadap lapisan keluaran, sehingga dapat mengurangi hilangnya fitur penting saat melakukan konvolusi. Sehingga perbedaan keduanya tidak besar hanya 0,39% untuk VGG-19 *random* maupun VGG-19 *grid*. Model yang cocok dalam percobaan penelitian ini terdapat pada VGG-19-SVM.

Pengembangan proses penelitian klasifikasi lukisan karya Van Gogh selanjutnya, yaitu dapat menggunakan *kernel* yang lain selain *kernel linear* dalam melakukan klasifikasi, seperti *kernel gaussian* (*Radial Basis Function*, RBF) untuk mencari nilai optimasi *random* dan *grid* dalam penelitian tersebut. Menambah jenis lukisan selain lukisan karya Van Gogh juga dapat dilakukan dalam penelitian selanjutnya, sehingga akan terbentuk *multi class* yang lebih beragam didalam penelitian nantinya. Selain itu, dapat juga menggunakan metode perhitungan optimasi yang lainnya untuk klasifikasi lukisan.

## DAFTAR PUSTAKA

- [1] Y. Rahmania, "Analisis Bahasa Rupa Pada Karya Lukis Mistaram Corak Kubistis Dekoratif Tahun 1998 - 2001," Universitas Brawijaya, 2017.
- [2] I. Muzdalifah and M. Marsudi, "Budi Bi; Teknik, Peran, dan Karya

- [3] Seni Lukis Cat Airnya," *J. Seni Rupa*, vol. 8, no. 3, pp. 35–48, 2020.
- [4] A. Lestari, "Kajian Karya Seni Lukis Vincent Van Gogh 'The Chairs,'" *J. Imaji*, vol. 3, no. 2, pp. 119–130, 2008.
- [5] A. K. Streit, "Ekspressionisme dan pengaruhnya terhadap desain poster Jerman pada sekitar masa perang dunia I," *J. RupaRupa*, vol. 2, no. 2, pp. 173–183, 2013.
- [6] J. Li, L. Yao, E. Hendriks, and J. Z. Wang, "Rhythmic Brushstrokes Distinguish van Gogh from His Contemporaries: Findings via Automated Brushstroke Extraction," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 6, pp. 1159–1176, 2012, doi: 10.1109/TPAMI.2011.203.
- [7] P. A. Octaviani, Y. Wilandari, and D. Ispriyanti, "Penerapan Metode Klasifikasi Support Vector Machine (SVM) pada Data Akreditasi Sekolah Dasar (SD) di Kabupaten Magelang," *J. Gaussian*, vol. 3, no. 4, pp. 811–820, 2014, doi: <https://doi.org/10.14710/j.gauss.v3i4.8092>.
- [8] Suyanto, *Machine Learning Tingkat Dasar dan Lanjut*. Bandung: Informatika, 2018.
- [9] W. H. Lopez Pinaya, S. Vieira, R. Garcia-Dias, and A. Mechelli, *Convolutional neural networks*. Elsevier Inc., 2019.
- [10] Y. Peng *et al.*, "CNN-SVM: A classification method for fruit fly image with the complex background," *IET Cyber-Physical Syst. Theory Appl.*, vol. 5, no. 4, pp. 1–5, 2020, doi: 10.1049/iet-cps.2019.0069.
- [11] X. X. Niu and C. Y. Suen, "A novel hybrid CNN-SVM classifier for recognizing handwritten digits," *Pattern Recognit.*, vol. 45, no. 4, pp. 1318–1325, 2012, doi: 10.1016/j.patcog.2011.09.021.
- [12] L. Hoang, S.-H. Lee, and K.-R. Kwon, "A 3D Shape Recognition Method Using Hybrid Deep Learning Network CNN-SVM," *Electronics*, vol. 9, no. 4, pp. 1–14, 2020, doi: 10.3390/electronics9040649.
- [13] J. Xu, L. Ma, W. Zhang, Q. Yang, X. Li, and S. Liu, "An Improved Hybrid CNN-SVM based Method for Bearing Fault Diagnosis Under Noisy Environment," *2019 Chinese Control Decis. Conf.*, pp. 4660–4665, 2019, doi: 10.1109/CCDC.2019.8832683.
- [14] X. Sun, L. Liu, C. Li, J. Yin, J. Zhao, and W. Si, "Classification for Remote Sensing Data With Improved CNN-SVM Method," *IEEE Access*, vol. 7, pp. 164507–164516, 2019, doi: 10.1109/ACCESS.2019.2952946.
- [15] X. Sun, J. Park, K. Kang, and J. Hur, "Novel hybrid CNN-SVM model for recognition of functional magnetic resonance images," *2017 IEEE Int. Conf. Syst. Man, Cybern.*, pp. 1001–1006, 2017, doi: 10.1109/SMC.2017.8122741.
- [16] A. F. M. Agarap, "An Architecture Combining Convolutional Neural Network (CNN) and Support Vector Machine (SVM) for Image Classification," *arXiv*, 2019.
- [17] F. H. Imai, M. R. Rosen, and R. S. Berns, "Multi-spectral imaging of van Gogh's Self-portrait at the National Gallery of Art, Washington, D.C.," *IS&T's 2001 PICS Conf. Proc.*, no. January, pp. 185–189, 2001.
- [18] G. Folego, O. Gomes, and A. Rocha, "From Impressionism to Expressionism: Automatically Identifying Van Gogh's Paintings," *2016 IEEE Int. Conf. Image Process.*, pp. 141–145, 2016, doi: 10.1109/icip.2016.7532335.
- [19] M. Al-Atoum, "The influence of vincent van gogh's religious background on his works in the nuene period," *Hamdard Islam. Q. J. Hamdard Natl. Found. Pakistan*, vol. 43, no. 3, pp. 151–173, 2020.
- [20] S. A. Centeno *et al.*, "Van Gogh's Irises and Roses: The contribution of chemical analyses and imaging to the assessment of color changes in the red lake pigments," *Herit. Sci.*, vol. 5, no. 1, pp. 1–11, 2017, doi: 10.1186/s40494-017-0131-8.
- [21] G. W. Retrani, "Van Gogh dari Imperssionisme menuju post Imperssionisme," *FIB UI*, p. 20, 2013.
- [22] M. Z. Alom *et al.*, "The History Began from AlexNet: A Comprehensive Survey on Deep Learning Approaches," *arXiv*, 2018.
- [23] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," *3rd Int. Conf. Learn. Represent. ICLR 2015*, pp. 1–14, 2015.
- [24] V. K. Chauhan, K. Dahiya, and A. Sharma, "Problem formulations and solvers in linear SVM: a review," *Artif. Intell. Rev.*, vol. 52, no. 2, pp. 803–855, 2019, doi: 10.1007/s10462-018-9614-6.

- [24] Meiriyama, "Klasifikasi Citra Buah Berbasis Fitur Warna HSV dengan Klasifikator SVM," *J. Komput. Terap.*, vol. 4, no. 1, pp. 50–61, 2018.
- [25] P. Wulandari, "Klasifikasi Tingkat Keganasan Kanker Serviks Menggunakan Metode Deep Residual Network (ResNet)," Universitas Islam Negeri Sunan Ampel Surabaya, 2019.
- [26] T. Nurhikmat, "Implementasi Deep Learning Untuk Image Classification Menggunakan Algoritma Convolutional Neural Network (CNN) Pada Citra Wayang Golek," Universitas Islam Indonesia, 2018.
- [27] R. Islamadina, F. Arnia, and K. Munadi, "Penggunaan Gray Level Co-Occurance Matrix Dari Koefisien Aproksimasi Wavelet untuk Deteksi Cacat Tekstil," *J. Buana Inform.*, vol. 6, no. 2, pp. 143–152, 2015, doi: 10.24002/jbi.v6i2.405.
- [28] M. Simon, E. Rodner, and J. Denzler, "ImageNet pre-trained models with batch normalization," *arXiv Prepr. arXiv1612.01452*, 2016.