SINTA Jurnal Sistem Informasi dan Teknologi Komputasi
Volume 2 Nomor 3 Juli 2025 [page: 153-162]
DOI: 10.61124/sinta.v2i3.92

ESTIMASI JARAK BERBASIS CNN dan REGRESI DALAM SISTEM DETEKSI KENDARAAN UDARA NIRAWAK (UAV) BERBASIS SUARA

Risa Farrid Christianti
Program Studi Teknik Elektro, Universitas Telkom Purwokerto, Indonesia
*e-mail: risabc@telkomuniversity.ac.id

Abstrak
Sistem deteksi UAV saat ini dilengkapi dengan berbagai jenis sensor sebagai alat deteksi, termasuk metode untuk menggabungkan data sensor. Namun, menggabungkan data dari beberapa sensor (multi-sensor data fusion, MDF) dapat menyebabkan kesalahan deteksi, yang antara lain disebabkan oleh noise lingkungan di sekitar sensor. Hal ini menghasilkan informasi peringatan yang salah (*False Alarm*) dari sistem. Penelitian ini bertujuan untuk mengembangkan sistem deteksi UAV berbasis suara menggunakan metode CNN dan Regresi, sehingga diperoleh informasi peringatan dini terkait ancaman UAV (drone), serta estimasi jarak terhadap larik node sensor yang lebih akurat. Suara UAV yang direkam oleh larik sensor audio diekstraksi untuk mendapatkan fitur data suara UAV. Ekstraksi fitur dilakukan dengan menggabungkan *Log-Mel Spectrogram* dan *Mel-Frequency Cepstrum Coefficient* (MFCC). Fitur ini digunakan untuk membangun model deteksi UAV sekaligus mengestimasi jarak menggunakan metode Convolutional Neural Network (CNN) dan Regresi. Eksperimen dilakukan menggunakan 5500 data primer berdasarkan 11 kelas jarak dalam satuan meter. Hasil eksperimen menunjukkan bahwa model deteksi UAV yang dikembangkan memiliki akurasi dan recall sebesar 91%, menandakan potensi tinggi dalam estimasi jarak UAV. Penelitian ini memberikan kontribusi penting bagi pengembangan ilmu di bidang pemrosesan sinyal akustik dan *deep learning* untuk deteksi objek udara. Secara praktis, sistem ini berpotensi diaplikasikan pada sistem keamanan, pemantauan wilayah, dan perlindungan infrastruktur kritis, khususnya di lingkungan dengan keterbatasan sensor visual atau radar.

Kata kunci: CNN dan Regresi, estimasi jarak UAV, fusi fitur MFCC dan Mel-Spectrogram, larik sensor audio.

Abstract
Current UAV detection systems are equipped with various types of sensors as detection tools, including methods for fusing sensor data. However, fusing data from multiple sensors (multi-sensor data fusion, MDF) can cause detection errors, which are caused by environmental noise around the sensor, among others. This results in false warning information (False Alarm) from the system. This study aims to develop a sound-based UAV detection system using CNN and Regression methods, so that early warning information regarding UAV (drone) threats is obtained, as well as a more accurate distance estimation to the sensor node array. UAV sounds recorded by the audio sensor array are extracted to obtain UAV sound data features. Feature extraction is carried out by combining Log-Mel Spectrogram and Mel-Frequency Cepstrum Coefficient (MFCC). This feature is used to build a UAV detection model while estimating the distance using Convolutional Neural Network (CNN) and Regression methods. Experiments were conducted using 5500 primary data points based on 11 distance classes in meters. Experimental results show that the developed UAV detection model has an accuracy and recall of 91%, indicating high potential in UAV distance estimation. This research provides important contributions to the development of acoustic signal processing and deep learning for aerial object detection. Practically, this system has potential applications in security systems, area monitoring, and critical infrastructure protection, particularly in environments with limited visual or radar sensors.

Keywords: Audio sensor array; CNN and Regression, MFCC and Mel-spectrogram feature fusion; UAV distance estimation.

1. PENDAHULUAN

Perkembangan pesat Kendaraan Udara Nirawak (UAV) telah merevolusi berbagai sektor, termasuk fotografi udara, pertanian, dan pemantauan keamanan. Namun, penggunaannya yang tidak sah telah menimbulkan kekhawatiran yang signifikan terkait keselamatan dan privasi. Sistem deteksi UAV konvensional, seperti radar, penglihatan, dan metode berbasis RF, telah banyak digunakan; namun, masing-masing memiliki keterbatasan dalam hal biaya, kompleksitas, dan sensitivitas lingkungan [1]. Deteksi berbasis akustik telah muncul sebagai alternatif yang menjanjikan karena sifatnya yang pasif dan kemampuannya untuk mendeteksi drone di luar garis pandang visual [2]. Metode pembelajaran mendalam, khususnya Jaringan Syaraf Tiruan Konvolusional (CNN), telah menunjukkan keberhasilan yang luar biasa dalam mempelajari pola suara dari data audio UAV, terutama ketika fitur seperti MFCC dan *mel-spectrogram* digunakan [3][4].

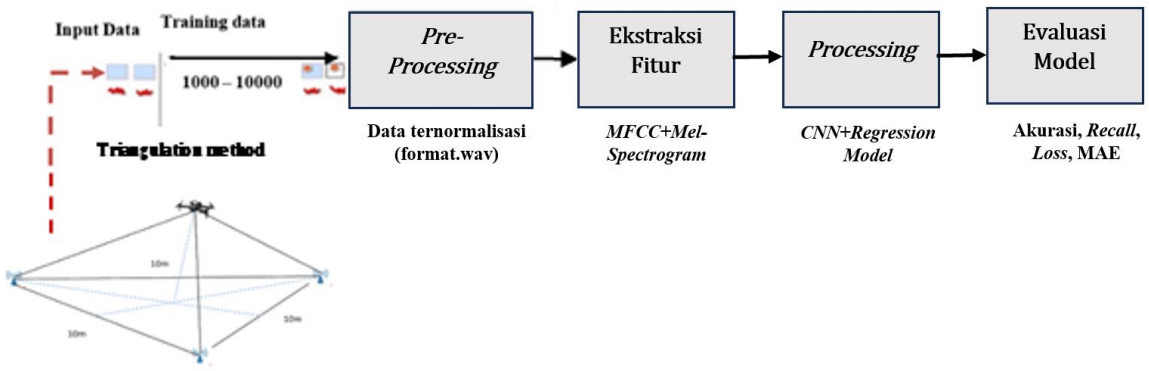
Metode deteksi drone mencakup beberapa modalitas, masing-masing dengan keterbatasannya. Sistem optik (kamera tampak atau inframerah) memanfaatkan algoritma penglihatan canggih (misalnya, deteksi objek YOLO), tetapi pada dasarnya membutuhkan garis pandang yang jelas dan cuaca yang mendukung; sistem ini gagal dalam kabut, kegelapan, atau di balik rintangan. Penganalisis frekuensi radio (RF) dapat mencegat sinyal komunikasi drone-ke-pengendali dalam jarak yang wajar; namun, sistem ini gagal mendeteksi drone otonom yang tidak memancarkan energi RF yang signifikan. Radar menawarkan pengawasan jarak jauh dan segala cuaca, tetapi biayanya tinggi dan kesulitan mendeteksi drone yang lambat atau kecil (karena algoritma peredam bising), menghambat penggunaannya untuk pemantauan drone di mana-mana [5][6][7][8][9][10][11].

Sebaliknya, deteksi UAV berbasis suara muncul sebagai pelengkap yang menarik, karena tidak bergantung pada visibilitas atau emisi RF, memungkinkan deteksi drone di lingkungan yang ramai, di malam hari, atau terhadap UAV yang senyap. Setiap drone dengan rotor menghasilkan tanda akustik yang khas (dengungan periodik dari bilah baling-baling dan motor) yang dapat dideteksi bahkan pada rasio sinyal terhadap bising yang rendah. Sensor akustik relatif murah dan pasif, sehingga memungkinkan aplikasi yang luas. Meskipun deteksi berbasis suara pada dasarnya terbatas jangkauannya oleh redaman atmosfer dan kebisingan sekitar, kemajuan dalam pemrosesan sinyal dan pengenalan pola semakin mengurangi masalah ini. Dengan mempertimbangkan hal ini, modalitas akustik merupakan cara yang menjanjikan untuk deteksi UAV, yang mendorong pendekatan berbasis audio [12][13][14][15]. Gap utama yang muncul adalah ketiadaan model yang mampu secara bersamaan melakukan klasifikasi keberadaan UAV dan estimasi jarak akurat menggunakan data akustik saja, terutama dalam lingkungan dengan noise tinggi.

Penelitian ini bertujuan untuk mengembangkan sistem deteksi UAV berbasis suara menggunakan kombinasi *Convolutional Neural Network* (CNN) dan regresi, yang mampu secara simultan melakukan klasifikasi keberadaan UAV serta mengestimasi jarak terhadap larik node sensor. Dengan menggabungkan fitur *Log-Mel Spectrogram* dan *Mel-Frequency Cepstral Coefficients* (MFCC) sebagai masukan model, sistem diharapkan dapat mengurangi tingkat *false alarm* yang disebabkan derau (*noise*) lingkungan, serta memberikan informasi peringatan dini yang lebih akurat untuk mendukung keamanan dan pemantauan wilayah.

2. METODE

Gambar 1 mengilustrasikan prinsip kerja keseluruhan sistem deteksi UAV untuk estimasi jarak menggunakan input data suara, yang dihasilkan dari penggabungan data dari tiga simpul sensor mikrofon. Data suara ditangkap dengan merata-ratakan intensitas suara dari setiap rekaman suara. Nilai rata-rata dari ketiga simpul sensor tersebut kemudian digunakan untuk pemrosesan dalam sistem.



Gambar 1. Blok diagram model usulan

Ketika sebuah UAV terbang dalam jangkauan, suaranya ditangkap sebagai sinyal deret waktu multi-kanal. Audio multi-kanal mentah tersebut terlebih dahulu melewati tahap pra-pemrosesan. Dataset primer untuk model klasifikasi, dikumpulkan melalui proses merekam, dan menyimpan data suara UAV, dari

beberapa tipe, namun dibatasi pada jenis multirotor (Quadrotor). Adapun tipe UAV yang digunakan antara lain: DJI Phantom 4 Pro, MJX Bugs 2, DJI Mavix Pro, dan DJI Spark. Dataset primer seluruhnya digunakan untuk melatih model identifikasi posisi, dengan pembagian kelas berdasarkan jarak rekam antara sumber suara UAV dengan sensor suara. Perekaman untuk setiap tipe quadrotor ini dilakukan dengan durasi 20 detik dan jarak tertentu (1m, 2m, 4m, 6m, 8m, 10m, 12m, 14m, 16m, 18m, dan 20m). Perangkat keras yang digunakan dalam melakukan perekaman suara UAV dan pengolahan data adalah: 3 modul sensor mic INMP441, 3 modul ESP32 Devkit VI, dan LCD I2C 16x2. Sedangkan perangkat lunaknya membutuhkan Arduino IDE, platform Google Colab dan Eagle.

Selanjutnya, dua set fitur diekstrak dari audio: MFCC dan spektrogram mel. Untuk ekstraksi MFCC, audio multikanal disegmentasi menjadi frame-frame pendek (dengan panjang frame tipikal 20–40 ms dan tumpang tindih 50%). Jendela *Hamming* diterapkan pada setiap frame untuk meruncingkan tepinya, lalu FFT dihitung untuk setiap frame guna mendapatkan spektrumnya. Spektrum dilewatkan melalui bank filter mel dan dikompresi logaritma, kemudian DCT menghasilkan MFCC. 40 koefisien MFCC pertama (tidak termasuk yang ke-0 jika mewakili energi DC) diekstrak, ditambah delta orde pertama dan kedua, membentuk vektor fitur per frame. Vektor MFCC ini dihitung untuk setiap kanal mikrofon. Secara paralel, spektrogram mel dihitung untuk audio setiap mikrofon. Hal ini melibatkan transformasi Fourier jangka pendek (STFT) dari sinyal (dengan panjang bingkai dan tumpang tindih yang sama), memetakan sumbu frekuensi ke skala mel, dan mengakumulasi kerapatan spektral daya dari waktu ke waktu ke dalam matriks 2D. Jendela waktu selama beberapa detik digunakan untuk menghasilkan *snapshot* spektogram mel yang berisi beberapa siklus rotasi bilah rotor. Untuk mengurangi rentang dinamis, skala logaritma atau dB digunakan untuk amplitudo spektogram. Setelah ekstraksi fitur, dilakukan juga fusi fitur di seluruh larik sensor. Ada dua tingkat fusi yang terjadi: fusi tingkat sensor (menggabungkan informasi dari beberapa mikrofon) dan fusi tingkat fitur (menggabungkan modalitas MFCC dan spektrogram).

Tujuan penggabungan fitur log-mel spectrogram dan MFCC adalah untuk memanfaatkan informasi waktu-frekuensi dan cepstral dari sinyal audio. Fitur log-mel Spectrogram dan MFCC dihitung untuk setiap kerangka waktu t, yang dinyatakan sebagai vektor fitur X, untuk fitur Log-Mel Spectrogram,

$$\boldsymbol{X}_{log-mel}(t) = [log - mel_1(t),\ log - mel_2(t), \dots, log - mel_M(t)]^T \quad (1)$$

dimana M adalah jumlah pita mel (dalam penelitian ini, n_mels = M = 128). Sedangkan vektor fitur MFCC,

$$\boldsymbol{X}_{MFCC}(t) = [MFCC_1(t),\ MFCC_2(t), \dots\dots, MFCC_K(t)]^T \quad (2)$$

dimana K adalah jumlah koefisien MFCC (dalam penelitian ini, n_mfcc = K = 40). Kedua vektor fitur menjadi satu vektor input:

$$\boldsymbol{X}(t) = [\boldsymbol{X}_{MFCC}(t) \oplus \boldsymbol{X}_{log-mel}(t)] \quad (3)$$

dimana $\oplus$ menunjukkan penggabungan, dan vektor yang dihasilkan memiliki ukuran K+M. Dalam rancangan penelitian ini, nilai K = 40 dan M = 128, maka X(t)∈ R^168 . Jadi, untuk setiap bingkai audio (frame), vektor fitur X(t) berisi koefisien cepstral (MFCC) dan energi mel-filterbank (log-mel), yang berukuran 168. Vektor fitur yang dihasilkan ini merepresentasikan karakteristik spektral dan temporal suara.

Matriks fitur untuk seluruh segmen audio dengan T frame, dimana T didapatkan dari perhitungan,

$$T = \frac{panjang\ sinyal\ audio(y)}{hop\_length} = \frac{10.000}{512} = 19{,}53\ \approx 20\ frame \quad (4)$$

menjadi matriks fitur X sebagai berikut:

$$\mathbf{X} = \begin{bmatrix} \mathbf{X}(1) \\ \mathbf{X}(2) \\ \vdots \\ \mathbf{X}(T) \end{bmatrix} \in \mathbb{R}^{T\times(K+M)} = \begin{bmatrix} \mathbf{X}(1) \\ \mathbf{X}(2) \\ \vdots \\ \mathbf{X}(T) \end{bmatrix} \in \mathbb{R}^{20\times168} = \begin{bmatrix} \mathbf{X}(1) \\ \mathbf{X}(2) \\ \vdots \\ \mathbf{X}(T) \end{bmatrix} \in \mathbb{R}^{3360} \quad (5)$$

Matriks ini akan berfungsi sebagai masukan ke sub-model MLP, dimana T adalah jumlah bingkai, dan setiap bingkai memiliki (K+M) fitur.

Pada pembentukan model, metode CNN digunakan dengan lapisan-lapisan tersembunyi dan fungsi-fungsi aktivasi yang telah ditentukan. Lapisan-lapisan tersebut mencakup lapisan yang terhubung

sepenuhnya, dan lapisan *Dropout* digunakan untuk mencegah model menjadi terlalu cocok dengan data pelatihan, yang dikenal sebagai *overfitting*. Lapisan keluaran menggunakan fungsi aktivasi, yang akan mengklasifikasikan fitur ke dalam sebelas kategori, yaitu "1m", "2m", "4m", "6m", "8m", "10m", "12m", "14m", "16m", "18m" dan "20m".

Evaluasi model CNN+Regresi pada penelitian ini dilakukan untuk mengukur kinerja sistem dalam mendeteksi UAV dan mengestimasi jaraknya secara akurat. Proses evaluasi dimulai dengan memuat bobot model terbaik hasil pelatihan sebelumnya, kemudian menyiapkan data uji yang terdiri dari dua target keluaran: (1) label kelas jarak untuk tugas klasifikasi, dan (2) nilai jarak numerik untuk tugas regresi. Data uji ini digunakan agar model diuji pada data yang belum pernah dilihat selama proses pelatihan, sehingga dapat mengukur kemampuan generalisasi. Tahap berikutnya adalah pengujian model terhadap data uji menggunakan dua metrik utama. Untuk tugas klasifikasi, digunakan metrik akurasi serta analisis presisi, *recall*, dan *f1-score* per kelas, yang juga divisualisasikan dalam bentuk *confusion matrix* untuk melihat distribusi prediksi benar dan salah pada setiap kelas jarak. Untuk tugas regresi, kinerja diukur menggunakan *Mean Absolute Error* (MAE), yang memberikan gambaran rata-rata selisih absolut antara jarak sebenarnya dan jarak hasil prediksi.

3. HASIL PENELITIAN

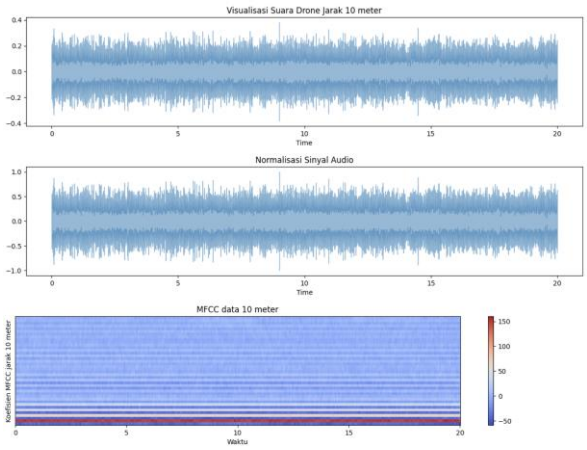
1. Tahap *Pre-Processing*

Dataset dimuat dari Google Drive dengan struktur folder berdasarkan kelas jarak UAV. Setiap file audio dinormalisasi untuk menyeragamkan level amplitudo, kemudian diekstraksi fitur MFCC sebanyak 40 koefisien dengan panjang frame tetap (max_frames=20) menggunakan *padding* atau *trimming*. Hasil ekstraksi dibentuk menjadi *array* (jumlah_data, 20, 40, 1) yang siap diproses CNN. Gambar 2 menunjukkan kelas dataset dalam folder di Google Drive berdasarkan kelas jarak (satuan: meter).

```
Detected class folders: ['20m', '2m', '6m', '4m', '8m', '1m', '14m', '18m', '16m', '12m', '10m']
```

Gambar 2. Hasil pembacaan dataset Google Drive dalam pemrograman Python

Gambar 3 menunjukkan visualisasi dari salah satu data dalam jarak tertentu (10 meter) dan hasil transformasinya menjadi fitur MFCC. Visualisasi pertama menampilkan bentuk gelombang audio mentah suara drone pada jarak 10 meter selama sekitar 20 detik. Amplitudo sinyal berkisar antara -0,35 hingga +0,35 dengan pola gelombang yang rapat dan konsisten, menandakan suara drone bersifat kontinu tanpa gangguan impulsif yang signifikan. Pada jarak ini, amplitudo telah sedikit melemah dibanding jarak dekat, namun masih cukup jelas terdeteksi. Visualisasi kedua memperlihatkan hasil normalisasi amplitudo dari sinyal tersebut, sehingga puncak amplitudo berada dalam rentang -1 hingga +1. Normalisasi ini bertujuan untuk menyamakan level volume di semua sampel audio agar perbedaan jarak atau kondisi perekaman tidak memengaruhi ekstraksi fitur, sekaligus meningkatkan stabilitas proses pelatihan CNN. Bentuk gelombangnya identik dengan sinyal mentah, namun dengan skala amplitudo yang sudah diseragamkan.



Gambar 3. Visualisasi data suara drone dan MFCC-nya

DOI: 10.61124/sinta.v2i3.92

Visualisasi ketiga adalah representasi *Mel-Frequency Cepstral Coefficients* (MFCC) dari sinyal suara drone pada jarak 10 meter, divisualisasikan dalam bentuk heatmap time-frequency. Sumbu horizontal menunjukkan waktu rekaman, sedangkan sumbu vertikal mewakili 40 koefisien MFCC yang memuat informasi energi dasar dan detail harmonik suara. Warna merah menandakan nilai MFCC tinggi atau energi dominan pada frekuensi tertentu, sementara biru menunjukkan nilai rendah atau negatif. Garis horizontal yang konsisten mencerminkan frekuensi harmonik stabil dari baling-baling drone, dengan warna merah pekat di bagian bawah menandakan dominasi energi pada frekuensi rendah yang khas. Intensitas warna yang relatif stabil sepanjang waktu menegaskan bahwa sinyal drone tetap kuat pada jarak 10 meter. Secara keseluruhan, pola MFCC yang jelas dan konsisten ini mengindikasikan kemudahan bagi model CNN dalam mengenali ciri khas suara drone, menjadikan data ini layak untuk dilanjutkan ke tahap training CNN+Regression.

2. Label dan Pemisahan Dataset

Label kelas jarak dikonversi menjadi *one-hot encoding* untuk klasifikasi, sementara target regresi (`y_reg`) diisi dengan nilai jarak numerik sebenarnya. Dataset dibagi menggunakan *stratified split* (test_size=0.3) agar distribusi kelas tetap seimbang pada train dan test set.

```
Input feature array shape: (5500, 20, 40, 1)
Encoded class labels: ['10m' '12m' '14m' '16m' '18m' '1m' '20m' '2m' '4m' '6m' '8m']
Label example - original: 20m encoded int: 6 one-hot: [0. 0. 0. 0. 0. 0. 1. 0. 0. 0. 0.]

Training samples: 3850 Testing samples: 1650
y_train one-hot shape: (3850, 11) y_test one-hot shape: (1650, 11)
```

Gambar 4. Visualisasi data suara drone dan MFCC-nya

Gambar 4 adalah bagian output dari proses pemberian label dan pemisahan dataset. Pada bagian *Input feature array shape*: (5500, 20, 40, 1), 5500 adalah total jumlah data sampel yang digunakan dalam eksperimen. 20 menyatakan panjang time frame dalam domain waktu setelah ekstraksi fitur (jumlah frame temporal). 40 menyatakan jumlah koefisien MFCC yang diekstraksi dari setiap frame (fitur spektral), dan 1 merupakan channel tunggal (karena audio mono, bukan stereo). Artinya, setiap data audio UAV telah dikonversi menjadi matriks 20x40 (time vs MFCC coefficient) dengan 1 channel, sehingga siap menjadi input layer CNN. Encoded Class Labels menunjukkan label jarak drone yang telah dikodekan. Dalam dataset yang disediakan, terdapat 11 kelas yang menyatakan jarak 1m hingga 20m sesuai kategori jarak sebenarnya. Label ini digunakan untuk *task* klasifikasi dari model CNN+Regression. Label Example memberikan contoh label yang dipetakan ke indeks tertentu pada larik label. Misalnya: label asli = "20m" encoded int = "6", artinya jarak 20m dipetakan ke indeks ke-6 pada array label. Dalam *One-hot encoding*, label direpresentasikan sebagai vektor dengan 11 elemen, di mana hanya satu posisi bernilai "1" sesuai jarak yang benar, sisanya bernilai "0". *One-hot* ini diperlukan agar output klasifikasi CNN bisa dibandingkan langsung dengan target pada *categorical crossentropy loss*.

Training samples (3850) menunjukkan bahwa data yang digunakan untuk melatih model (sekitar 70% dari total data). Testing samples (1650) menunjukkan bahwa data yang digunakan untuk menguji kinerja model setelah training (sekitar 30%). Pemisahan ini dilakukan menggunakan *Stratified Split*, sehingga distribusi kelas pada *train* dan *test* tetap seimbang. Bentuk Label Training dan Testing-nya dinyatakan dalam "y_train one-hot shape: (3850, 11)" dan "y_test one-hot shape: (1650, 11)". Setiap sampel train dan test memiliki vektor label berukuran 11 (sesuai jumlah kelas jarak UAV). Bentuk (3850, 11) dan (1650, 11) berarti ada masing-masing 3850 dan 1650 vektor label, dengan 11 dimensi tiap vektor. Dari hasil pemrograman ini dapat disimpulkan bahwa data UAV sudah berhasil di-preprocessing (*MFCC extraction + reshaping*) menjadi format input CNN. Label jarak sudah di-encode ke format *one-hot* untuk klasifikasi. Dataset sudah dipisahkan dengan proporsi 70% training dan 30% testing dengan distribusi kelas yang seimbang. Data siap untuk masuk ke tahap model *training* CNN+Regresi.

3. Arsitektur Model CNN+Regresi

Model memiliki *backbone* CNN yang sama untuk ekstraksi fitur, lalu bercabang menjadi dua output: klasifikasi (softmax) untuk memprediksi kelas jarak, dan regresi (linear) untuk memprediksi jarak kontinu.

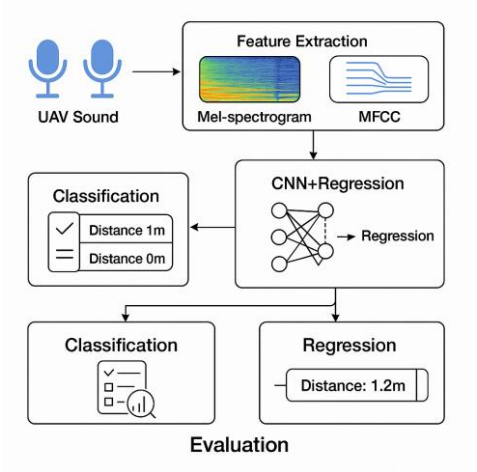
Pendekatan ini memungkinkan kedua tugas saling berbagi representasi fitur yang dipelajari. Adapun arsitektur model CNN+Regresi dapat direpresentasikan dalam Tabel 1 sebagai berikut:

Tabel 1. Arsitektur model CNN+Regression

Layer (Tipe)	Bentuk Keluaran	Param #
Input_layer_1 (InputLayer)	(None, 20, 40, 1)	0
Conv2d_2 (Conv2D)	(None, 18, 38, 32)	320
Max_pooling2d_2 (Maxpooling2D)	(None, 9, 19, 32)	0
Conv2d_3 (Conv2D)	(None, 7, 17, 64)	18.496
Max_pooling2d_3 (MaxPooling2D)	(None, 3, 8, 64)	0
Flatten_1 (Flatten)	(None, 1536)	0
Dense_2 (Dense)	(None, 128)	196.736
Classification_output (Dense)	(None, 11)	1.419
Regression_output (RegessionCalibration)	(None, 1)	0

Total params: 216.971 (847,54 KB)
Trainable params: 216.971 (847,54 KB)
Non-trainable params: 0 (0,00 B)

Model CNN+Regresi pada Tabel 1 dirancang untuk melakukan dua tugas sekaligus, yaitu klasifikasi dan estimasi numerik (regresi) dari fitur yang diekstraksi. Arsitektur diawali dengan 'InputLayer' berukuran (20, 40, 1) yang merepresentasikan citra atau matriks fitur 2D berskala abu-abu. Data kemudian diproses oleh dua blok 'Conv2D' dan 'MaxPooling2D': lapisan konvolusi pertama memiliki 32 filter berukuran kernel 3×3 menghasilkan output (18, 38, 32), diikuti pooling yang mereduksi ukuran menjadi (9, 19, 32); lapisan konvolusi kedua memiliki 64 filter kernel 3×3 menghasilkan (7, 17, 64), lalu pooling menjadi (3, 8, 64). Hasilnya kemudian di-*flatten* menjadi vektor 1D berukuran 1.536 elemen, lalu diproses oleh 'Dense layer' berukuran 128 neuron sebagai lapisan *fully connected*. Output dari lapisan ini bercabang ke dua jalur: 'classification_output' berupa Dense layer dengan 11 neuron (menghasilkan 11 kelas) dan 'regression_output' berupa 'RegressionCalibration' dengan 1 output untuk estimasi nilai kontinu, misalnya jarak UAV. Total parameter yang dilatih sebanyak 216.971 parameter, menunjukkan model relatif ringan namun cukup kompleks untuk mengekstraksi fitur spasial sekaligus melakukan *multi-task learning*. Gambar 5 menggambarkan diagram hubungan alur kerja CNN+Regresi sebagai berikut:



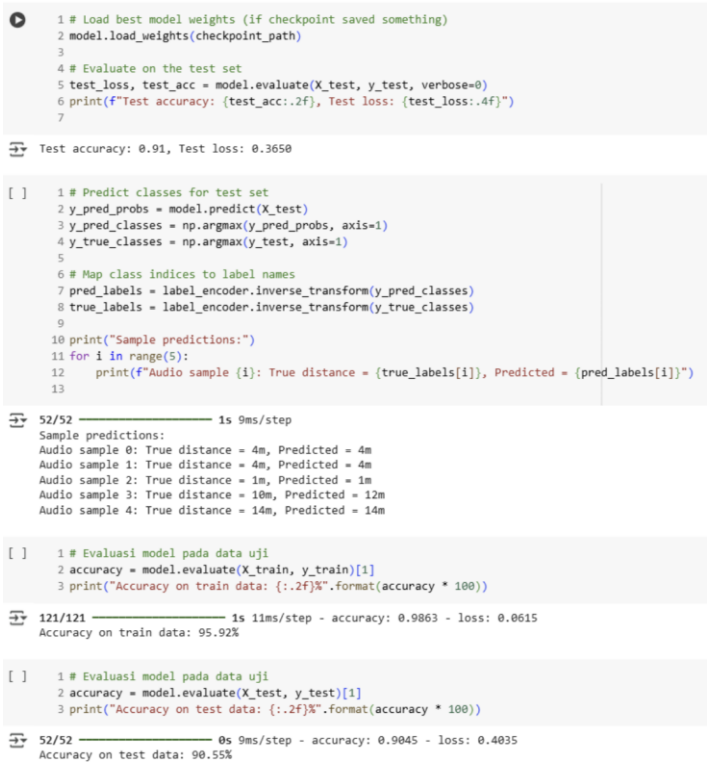
Gambar 5. Hubungan alur kerja CNN+Regresi

Diagram alur tersebut menggambarkan proses lengkap estimasi jarak UAV berbasis suara menggunakan kombinasi metode CNN+Regresi. Proses dimulai dari perekaman suara UAV melalui mikrofon yang kemudian diproses pada tahap *feature extraction*. Pada tahap ini, sinyal audio diubah menjadi representasi fitur menggunakan dua teknik utama, yaitu *Mel-spectrogram* dan MFCC, yang berfungsi untuk menangkap ciri khas spektral dan temporal dari suara UAV. Fitur-fitur ini kemudian menjadi masukan bagi model CNN+Regresi, yang bertugas melakukan dua jenis keluaran secara bersamaan.

Pertama, model melakukan klasifikasi untuk mengidentifikasi kategori jarak tertentu (misalnya 1 m, 2 m, atau 10 m) berdasarkan pola audio yang dikenali. Kedua, model juga melakukan regresi untuk memprediksi nilai jarak secara numerik yang lebih presisi, misalnya 1,2 m, sehingga mampu memberikan estimasi yang lebih detail. Hasil dari kedua keluaran tersebut kemudian masuk ke tahap evaluasi, di mana kinerja model diukur untuk memastikan akurasi klasifikasi dan ketepatan prediksi regresi. Dengan demikian, diagram ini memperlihatkan integrasi dua pendekatan—klasifikasi dan regresi—yang saling melengkapi, sehingga sistem tidak hanya mampu mengelompokkan jarak UAV dalam kelas-kelas tertentu, tetapi juga memberikan estimasi jarak yang akurat dalam bentuk nilai kontinyu, yang sangat bermanfaat untuk aplikasi pemantauan dan pelacakan UAV secara *real-time*. Pada proses pelatihan dataset, nilai batch size diatur pada nilai batch_size = 32, artinya, model akan memproses 32 sampel sekaligus dalam satu iterasi sebelum melakukan pembaruan bobot. Jumlah epoch diatur pada num_epochs = 100, dimana seluruh dataset pelatihan akan diproses sebanyak 100 kali secara penuh. Nilai learning rate mengikuti pengaturan default dari Optimizer Adam (default = 0,001).

4. Evaluasi Hasil

Hasil pemodelan CNN+Regresi yang ditampilkan pada gambar 6 menunjukkan bahwa model telah dilatih dan diuji dengan performa yang sangat baik. Setelah memuat bobot terbaik dari *checkpoint*, evaluasi pada *test set* menghasilkan akurasi 91% dengan *loss* 0,3650, yang menunjukkan kemampuan model untuk mengklasifikasikan jarak UAV dari data audio dengan tingkat kesalahan relatif rendah. Prediksi contoh menunjukkan bahwa model mampu memprediksi jarak yang benar pada semua sample yang diuji (misalnya 4m → 4m, 1m → 1m, 10m → 12m, dan 14m → 14m), walaupun ada satu kasus jarak 10m yang terprediksi 12m, yang menunjukkan adanya sedikit deviasi pada kelas yang jaraknya berdekatan. Evaluasi pada *training set* menghasilkan akurasi 95,92% dengan *loss* 0,0615, sedangkan evaluasi pada *test set* mencapai akurasi 90,55% dengan loss 0,4035. Perbedaan kecil antara akurasi pelatihan dan pengujian menunjukkan bahwa model memiliki kemampuan generalisasi yang baik, meskipun ada sedikit indikasi *overfitting* karena akurasi pelatihan lebih tinggi. Tingkat akurasi di atas 90% pada data uji mengindikasikan bahwa CNN yang digunakan cukup efektif dalam mengekstraksi pola dari *spectrogram* audio, sementara komponen regresi membantu memperkirakan jarak UAV secara lebih presisi.



Gambar 6. Hasil pemodelan CNN+Regresi

DOI: 10.61124/sinta.v2i3.92

Kinerja klasifikasi diukur dengan akurasi, presisi, *recall*, *F1-score*, dan *confusion matrix*, sedangkan kinerja regresi diukur dengan *Mean Absolut Error* (MAE). Hasil yang ditunjukkan pada Gambar 7 menunjukkan bahwa akurasi tinggi pada jarak dekat karena sinyal UAV lebih jelas, namun menurun pada jarak jauh akibat penurunan SNR dan kemiripan pola fitur antar kelas jarak yang berdekatan.

	precision	recall	f1-score	support
10m	0.91	0.92	0.91	150
12m	0.88	0.87	0.88	150
14m	0.97	0.92	0.95	150
16m	0.80	0.91	0.85	150
18m	0.78	0.90	0.84	150
1m	0.91	0.96	0.94	150
20m	0.94	0.89	0.91	150
2m	0.95	0.92	0.93	150
4m	0.91	0.97	0.94	150
6m	0.97	0.97	0.97	150
8m	0.97	0.75	0.85	150
accuracy			0.91	1650
macro avg	0.91	0.91	0.91	1650
weighted avg	0.91	0.91	0.91	1650

(a)

```
Confusion Matrix:
 [[138   6   0   4   1   0   0   0   0   0   1]
 [  8 130   1   4   5   0   1   0   1   0   0]
 [  1   2 138   2   3   0   4   0   0   0   0]
 [  1   0   0 136   9   2   2   0   0   0   0]
 [  0   2   2   9 135   0   1   0   0   0   1]
 [  0   0   0   1   0 144   0   4   1   0   0]
 [  0   0   0   4  13   0 133   0   0   0   0]
 [  0   2   0   1   0   6   0 138   1   2   0]
 [  1   1   0   0   0   0   0   2 145   1   0]
 [  0   0   0   0   0   0   0   2   2 145   1]
 [  3   4   1   8   6   6   0   0   9   1 112]]
```

(b)

Gambar 7. (a)Metrik kinerja model; (b) *Confusion Matrix* hasil evaluasi

Hasil kinerja model CNN+Regresi yang ditunjukkan pada Gambar 7(a), dimana nilai akurasi keseluruhan sebesar 91%, dengan *macro average* dan *weighted average* yang konsisten di angka 0,91 untuk *precision, recall,* dan *F1-score*. Hal ini menunjukkan bahwa model memiliki performa yang seimbang dalam mendeteksi seluruh kelas jarak UAV, tanpa bias yang signifikan terhadap kelas tertentu. Pada sebagian besar kelas jarak, seperti 1m, 2m, 4m, 6m, dan 14m, nilai *precision, recall,* dan *F1-score* berada di atas 0,90, yang menandakan tingkat ketepatan prediksi dan cakupan yang baik. Namun, terdapat kelas seperti 16m, 18m, dan 8m yang performanya relatif lebih rendah, khususnya pada *recall* untuk 8m (0,75), yang menunjukkan model cenderung melewatkan beberapa *instance* dari kelas ini. *Confusion Matrix* pada Gambar 7(b), memberikan gambaran detail tentang distribusi prediksi model. Terlihat bahwa diagonal matriks sebagian besar memiliki angka tinggi (prediksi benar), menandakan akurasi tinggi. Namun, terdapat kesalahan prediksi antarjarak yang berdekatan, seperti antara 16m dan 18m, atau antara 8m dengan jarak lain yang dekat. Hal ini kemungkinan disebabkan oleh kemiripan pola spektrum suara UAV pada jarak yang mirip, sehingga model lebih sulit membedakannya. Prediksi sampel menunjukkan bahwa model mampu memprediksi jarak dengan tepat pada beberapa kasus (misalnya 4m → 4m, 1m → 1m, 14m → 14m), tetapi masih terdapat peluang perbedaan pada data lain yang tidak ditampilkan. Gambar 8 menunjukkan hasil evaluasi regresi dari model CNN+Regresi yang ditunjukkan sebagai berikut:

```
Overall Test Loss: 1.7015
Classification Test Loss: 0.3688
Regression Test Loss: 4.4245
Classification Test Accuracy: 0.91
Regression Test MAE: 0.5789

52/52 ━━━━━━━━━━━━━━━━━━━━ 1s 9ms/step
Sample predictions:
Audio sample 0: True distance = 4m, Predicted = 4m
Audio sample 1: True distance = 4m, Predicted = 4m
Audio sample 2: True distance = 1m, Predicted = 1m
Audio sample 3: True distance = 10m, Predicted = 10m
Audio sample 4: True distance = 14m, Predicted = 14m
```

Gambar 8. Hasil evaluasi model CNN+Regresi dengan MAE

Model ini memiliki dua output, yaitu Klasifikasi untuk memprediksi kelas jarak UAV (kategori: 1m, 2m, 4m, dst.), dan Regresi untuk memperkirakan jarak UAV secara numerik (nilai kontinyu dalam meter). Dari hasil evaluasi:

1. *Overall Test Loss* = 1,7015 → gabungan dari *loss* klasifikasi dan regresi.
2. *Classification Test Loss* = 0,3688 → tingkat *error* model klasifikasi relatif rendah.
3. *Regression Test Loss* = 4,4245 → menunjukkan *error* rata-rata model regresi terhadap target numerik.
4. *Classification Test Accuracy* = 0,91 → model benar mengklasifikasikan 91% sampel test.

DOI: 10.61124/sinta.v2i3.92

5. *Regression Test* MAE = 0,5789 → rata-rata selisih absolut prediksi jarak terhadap jarak sebenarnya sekitar 0,58 meter.

4. DISKUSI

Arsitektur CNN+Regresi ini tergolong ringan dan efisien, cocok untuk dataset UAV berbasis MFCC dan *mel-spectrogram* berukuran kecil hingga menengah. Dengan hanya dua lapisan konvolusi, model ini mampu mengekstraksi fitur dasar namun mungkin memiliki keterbatasan dalam menangkap pola kompleks pada data dengan *noise* tinggi. Keberadaan *Dropout* membantu menjaga generalisasi, sedangkan jumlah parameter yang tidak terlalu besar mempermudah implementasi di perangkat dengan keterbatasan komputasi. Model ini layak untuk baseline CNN+Regresi pada UAV sound-based distance estimation karena ringan, mudah dilatih, dan cepat dieksekusi. Namun, untuk peningkatan akurasi dan ketahanan terhadap noise, dapat dipertimbangkan penambahan jumlah layer konvolusi, penggunaan batch normalization, atau arsitektur yang lebih dalam seperti VGG-like CNN atau ResNet varian ringan.

Arsitektur CNN+Regresi ini juga berhasil menggabungkan kekuatan klasifikasi dan prediksi jarak dari data audio UAV secara efektif. Tingkat akurasi tinggi di kedua dataset menunjukkan model mampu mengenali pola akustik jarak UAV dengan baik. Namun, adanya prediksi jarak yang sedikit meleset pada kelas berdekatan mengindikasikan bahwa perbedaan fitur antara jarak yang mirip mungkin tidak terlalu signifikan sehingga masih memerlukan optimalisasi, misalnya dengan penambahan data, fine-tuning hiperparameter, atau penggunaan teknik augmentasi data audio yang lebih bervariasi. Hasil ini menunjukkan bahwa model CNN+Regresi cukup baik dalam mengklasifikasikan jarak UAV ke dalam kelas-kelas yang sudah didefinisikan, dengan tingkat akurasi yang tinggi (91%). Loss klasifikasinya yang rendah (0,3688) menandakan kemampuan generalisasi yang baik.

Namun, pada sisi regresi, nilai MAE = 0,5789m sebenarnya tergolong akurat untuk estimasi jarak secara numerik, tetapi *Regression Loss* = 4,4245 cukup besar. Hal ini bisa disebabkan oleh beberapa faktor: (1) perbedaan skala target regresi dengan target klasifikasi; (2) adanya outlier atau noise pada data audio UAV; (3) Model lebih "fokus" pada klasifikasi ketimbang regresi karena loss klasifikasi lebih dominan di *training*. Jika tujuan utama adalah mendapatkan estimasi jarak kontinyu yang lebih presisi, perlu dilakukan *tuning* bobot *loss weighting* agar regresi mendapat porsi pembelajaran lebih besar. Alasan penggunaan MAE dalam evaluasi model CNN+Regresi ini adalah karena:

1. Kesederhanaan Interpretasi: MAE memberikan nilai rata-rata dari selisih absolut antara prediksi dan nilai sebenarnya, yang langsung dapat diinterpretasikan dalam satuan asli target. Dalam kasus model CNN+Regresi ini, targetnya adalah jarak UAV dalam meter. Contohnya, MAE = 0,5789 berarti rata-rata kesalahan prediksi jarak adalah sekitar 0,58 meter, yang mudah dipahami secara praktis.
2. Tidak Terlalu Sensitif terhadap Outlier: MAE tidak mengkuadratkan selisih error (berbeda dengan MSE), sehingga tidak terlalu terpengaruh oleh error besar yang ekstrem. Hal ini penting untuk data jarak UAV, karena terkadang ada sedikit prediksi yang meleset jauh akibat *noise*, namun tidak boleh mendominasi perhitungan *error* keseluruhan.
3. Cocok untuk Evaluasi Kinerja Regresi Multitugas: Model CNN+Regresi ini mempunyai dua output, yaitu Klasifikasi (memprediksi label jarak dalam kategori seperti 4m, 6m, dst.) dan Regresi (memprediksi jarak numerik secara langsung). MAE digunakan di regresi karena pengguna ingin tahu seberapa jauh rata-rata jarak prediksi dari jarak sebenarnya, tanpa membedakan apakah kesalahan itu positif atau negatif.
4. Mendukung Analisis Kinerja Secara Praktis di Lapangan: Dalam skenario UAV *detection*, informasi berapa meter rata-rata kesalahan prediksi jauh lebih berguna secara operasional, dibanding nilai error kuadrat atau log loss. Hal ini membantu tim lapangan memahami toleransi error yang bisa diterima untuk sistem deteksi.

Model CNN+Regresi yang digunakan mampu melakukan klasifikasi jarak UAV dengan akurasi tinggi (91%) dan kesalahan rata-rata prediksi jarak kontinu hanya sekitar 0,58 meter. Hasil ini sudah cukup baik untuk sistem identifikasi posisi UAV berbasis audio. Namun, performa regresi masih bisa ditingkatkan dengan menyeimbangkan kontribusi *loss*, melakukan normalisasi target regresi, dan memperbanyak variasi data *training* untuk mengurangi *overfitting* pada klasifikasi.

5. KESIMPULAN

Penelitian ini berhasil mengembangkan model CNN+Regresi untuk estimasi jarak UAV berbasis audio dengan capaian akurasi tinggi, yaitu 91% pada berbagai skenario jarak. Keunggulan model terletak pada kemampuannya mempertahankan kinerja optimal pada jarak pendek hingga menengah (1m–14m), menunjukkan efektivitas ekstraksi fitur suara dalam membedakan jarak UAV. Kontribusi penelitian ini adalah memberikan pendekatan yang lebih akurat dan andal untuk sistem deteksi serta estimasi jarak UAV berbasis sensor audio, yang dapat memperkuat sistem peringatan dini pada bidang keamanan dan pemantauan. Potensi pengembangan ke depan mencakup peningkatan akurasi pada jarak yang sulit dibedakan melalui strategi augmentasi data dan pengayaan arsitektur model.

UCAPAN TERIMA KASIH

Terima kasih kepada Telkom University Purwokerto yang telah memberikan ijin kegiatan penelitian, fasilitas dan kesempatan untuk mengembangkan topik riset ini, serta beberapa pihak lain yang turut menyukseskan kegiatan penelitian ini.

DAFTAR PUSTAKA

[1] Q. Dong, Y. Liu, and X. Liu, "Drone sound detection system based on feature result-level fusion using deep learning," *Multimed. Tools Appl.*, vol. 82, no. 1, pp. 149–171, 2023, doi: 10.1007/s11042-022-12964-3.

[2] J. Zhong, A. Fan, K. Fan, W. Pan, and L. Zeng, "Research on the UAV Sound Recognition Method Based on Frequency Band Feature Extraction," *Drones*, vol. 9, no. 5, pp. 1–14, 2025, doi: 10.3390/drones9050351.

[3] D. Tejera-Berengue, F. Zhu-Zhou, M. Utrilla-Manso, R. Gil-Pita, and M. Rosa-Zurera, "Analysis of Distance and Environmental Impact on UAV Acoustic Detection," *Electronics*, vol. 13, no. 3. 2024. doi: 10.3390/electronics13030643.

[4] Y. Wang, F. Fagiani, K. Ho, and E. Matson, "A Feature Engineering Focused System for Acoustic UAV Payload Detection," *Proc. 14th Int. Conf. Agents Artif. Intell. (ICAART 2022)*, vol. 3, no. Icaart, pp. 470–475, 2022, doi: 10.5220/0010843800003116.

[5] P. Wellig *et al.*, "Radar systems and challenges for C-UAV," *Proc. Int. Radar Symp.*, vol. 2018-June, pp. 1–8, 2018, doi: 10.23919/IRS.2018.8448071.

[6] P. Nguyen, H. Truong, and M. Ravindranathan, "Passive Rf-Based Drone," vol. 21, no. 4, pp. 30–34, 2017.

[7] M. S. Allahham, T. Khattab, and A. Mohamed, "Deep Learning for RF-Based Drone Detection and Identification: A Multi-Channel 1-D Convolutional Neural Networks Approach," *2020 IEEE Int. Conf. Informatics, IoT, Enabling Technol. ICIoT 2020*, pp. 112–117, 2020, doi: 10.1109/ICIoT48696.2020.9089657.

[8] O. O. Medaiyese, A. Syed, and A. P. Lauf, "Machine Learning Framework for RF-Based Drone Detection and Identification System," 2020, [Online]. Available: http://arxiv.org/abs/2003.02656

[9] M. F. Al-Sa'd, A. Al-Ali, A. Mohamed, T. Khattab, and A. Erbad, "RF-based drone detection and identification using deep learning approaches: An initiative towards a large open source drone database," *Futur. Gener. Comput. Syst.*, vol. 100, pp. 86–97, 2019, doi: 10.1016/j.future.2019.05.007.

[10] M. Messina and G. Pinelli, *Classification of Drones with a Surveillance Radar Signal*, vol. 11754 LNCS. 2019. doi: 10.1007/978-3-030-34995-0_66.

[11] J. Busset *et al.*, "Detection and tracking of drones using advanced acoustic cameras," *Unmanned/Unattended Sensors Sens. Networks XI; Adv. Free. Opt. Commun. Tech. Appl.*, vol. 9647, p. 96470F, 2015, doi: 10.1117/12.2194309.

[12] T. P. Banerjee and S. Das, "Multi-sensor data fusion using support vector machine for motor fault detection," *Inf. Sci. (Ny).*, vol. 217, pp. 96–107, 2012, doi: 10.1016/j.ins.2012.06.016.

[13] S. Jeon, J. W. Shin, Y. J. Lee, W. H. Kim, Y. H. Kwon, and H. Y. Yang, "Empirical study of drone sound detection in real-life environment with deep neural networks," *25th Eur. Signal Process. Conf. EUSIPCO 2017*, vol. 2017-Janua, pp. 1858–1862, 2017, doi: 10.23919/EUSIPCO.2017.8081531.

[14] J. Guo, I. Ahmad, and K. H. Chang, "Classification, positioning, and tracking of drones by HMM using acoustic circular microphone array beamforming," *Eurasip J. Wirel. Commun. Netw.*, vol. 2020, no. 1, 2020, doi: 10.1186/s13638-019-1632-9.

[15] J. Ngiam, A. Khosla, M. Kim, J. Nam, H. Lee, and A. Y. Ng, "Multimodal deep learning," in *ICML*, 2011.