nodeflux

Extending Vision Beyond Imagination

# Internship Report

Eufrat Tsaqib Qasthari

Pembimbing

Ilham Imaduddin

# Outline

1. Pendahuluan
2. Aplikasi Nodeflux
3. Teknologi Nodeflux
4. Proyek Penelitian
5. Proyek Penelitian: Streaming
6. Proyek Penelitian: Inferencing
7. Hasil penelitian

# Pendahuluan

Nodeflux:

- Perusahaan berbasis kecerdasan buatan (AI) untuk melakukan analisis video dari instrumen kamera konvensional.
- Perusahaan pertama di Indonesia yang bergerak di bidang Intelligence Video Analytics (IVA).
- Satu-satunya perusahaan di Indonesia yang masuk ke NVIDIA Inception Program (Akselerator untuk AI dan Deep Learning).
- Perusahaan teknologi rintisan baru (startup), dirintis 2016.
- Sudah digunakan oleh Kepolisian RI, Badan Intelijen Negara (BIN), Jasa Marga, Transjakarta, Jakarta Smart City, Bandung Command Center, GOJEK dll

# Aplikasi Nodeflux

Tribuntechno  Kamis, 8 November 2018  Cari  Netw

Home  News  Gadget  App

## Teknologi Face Recognition Nodeflux Amankan Pertemuan Tahunan IMF-World Bank di Bali

Rabu, 17 Oktober 2018 09:11 WIB

METROTV NEWS.COM

HOME  PROGRAM  POPULAR VIDEO

## Nodeflux, Aplikasi Pendeteksi Objek yang Tertangkap CCTV

06 Januari 2018 17:22 WIB  526

CNN Indonesia  Home  Nasional  Internasional  Ekonomi  Olahraga  Teknologi  Hiburan  Gaya Hi

Jokowi juga mencoba *face recognition* Nodeflux. Nama dan jabatan Jokowi langsung terlihat ketika wajahnya terdeteksi kamera. Sistem pengenal wajah yang terkoneksi dengan data penduduk dan catatan sipil alias dukcapil ini sudah digunakan sejak penyelenggaraan Asian Games. **(chri/end)**

detikinet

HOME  BERITA  GADGET  TIPS & TRICK  GAMES  KLINIK INET  FOTOSTOP  NGOPI  INET GRAFI

Most Popular · Inet Highlights

Home / Cyber Life / Detail berita  FOLLOW detikInet

Rabu, 27 Jun 2018 16:36 WIB

## Atur Arus Mudik, Jasa Marga Pakai Teknologi AI

Achmad Rouzni Noor II - detikInet

# Teknologi Nodeflux

Teknologi:

- Sumber video dari kamera CCTV.
- Analisis video berbasis jaringan syaraf tiruan (Artificial Neural Network) dengan ratusan layer (Deep Neural Network) yang dikovolusikan (Convolutional Neural Network).
- Model AI dilatih dan diinferensikan pada NVIDIA GPU (Graphics Processing Unit) .
- Model AI dapat diinferensikan secara offline ataupun online.
- Hasil analisis ditampilkan di halaman web.

# Intelligent Video Analytics

### Face Recognition

An accurate and high performance facial searching and recognition using deep learning algorithm

### License Plate Recognition

Developed to adapt real condition in Indonesia like unstandardized plate, Nodeflux LPR can learn over the time to improve the recognition ability.

### Flood Monitoring

Minimize factor that can cause flood in your city by detecting water level, unwanted object presence, or trash.

### Heatmap

Visualizing overall activity level within an area, to indicate hot spots, according to where people stop at often or for longer periods of time.

### Crowd Analytic

Nodeflux Crowd Behavior Analytic provide an accurate and real-time abnormal crowd behaviour warning system.

### Trash Detection

Detect any object and track their movement on predefined region of interest.

### Traffic Measurement

Optimize traffic management by measuring vehicle demography, vehicle count, vehicle density.

### People Flow

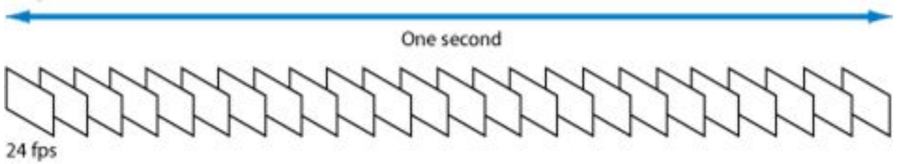
Tracks the people's path within an area. This helps you to identify the dominant paths taken by people within a designated time frame.

### Product Ranking

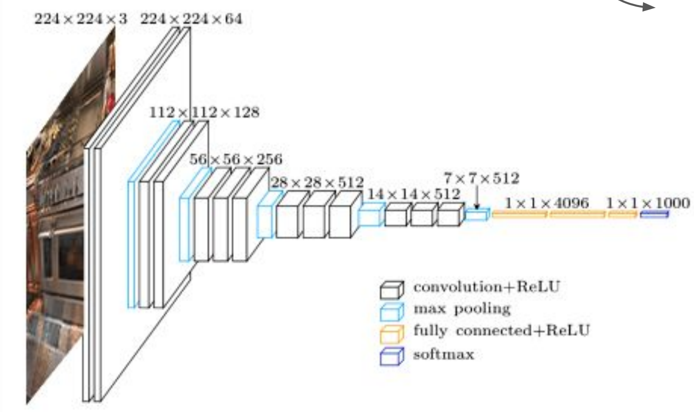
Use people dwelling spot and time to obtain additional information such as product ranking. Analyze which product that can attract people more.

# Teknologi Nodeflux

2. Video dipecah menjadi gambar




1. Video direkam kamera CCTV




3. Gambar dianalisis dengan model Deep Learning.

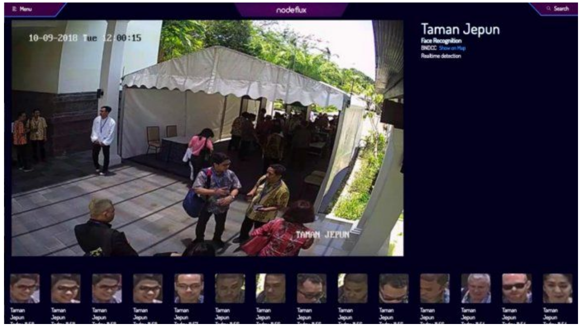

4. Hasil analisis ditampilkan di web.

# Teknologi Nodeflux

Analisis offline


Analisis video dengan GPU
(NVIDIA Tesla P4/P40)

Output video
Sudah dianalisis

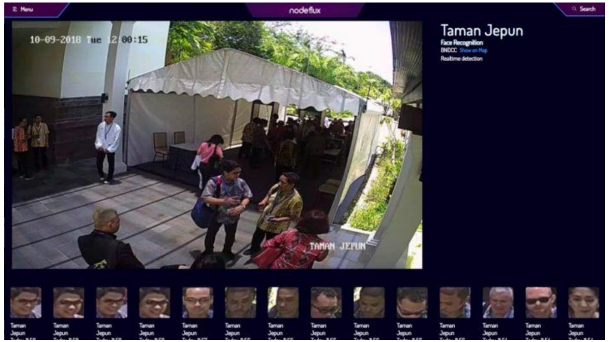


# Teknologi Nodeflux

Cloud
servers

Output video
belum dianalisis

Output video
sudah dianalisis



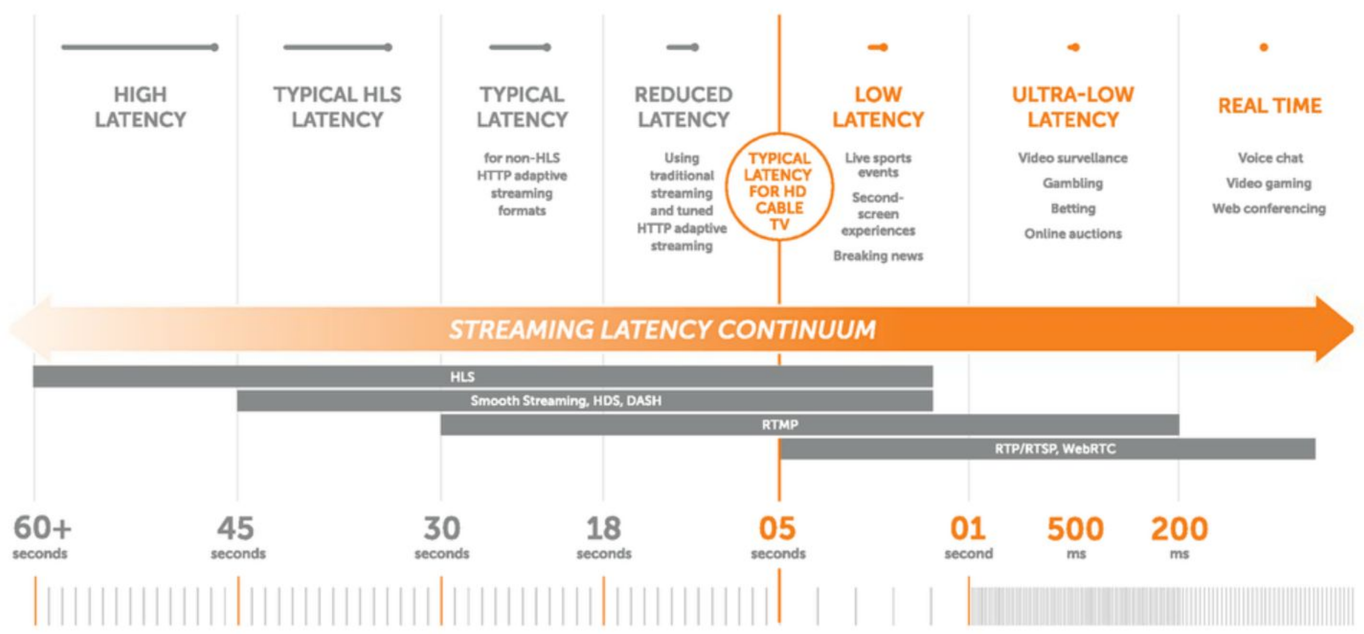
# Proyek Penelitian

Object Detection Inferencing Streaming
Berbasis Artificial Neural Network

Streaming
(mengalirkan data)

Inferencing
(mendapatkan hasil analisis dari data)

# Proyek Penelitian: Streaming




Protokol

- DASH
- RTP/RTSP
- WebRTC

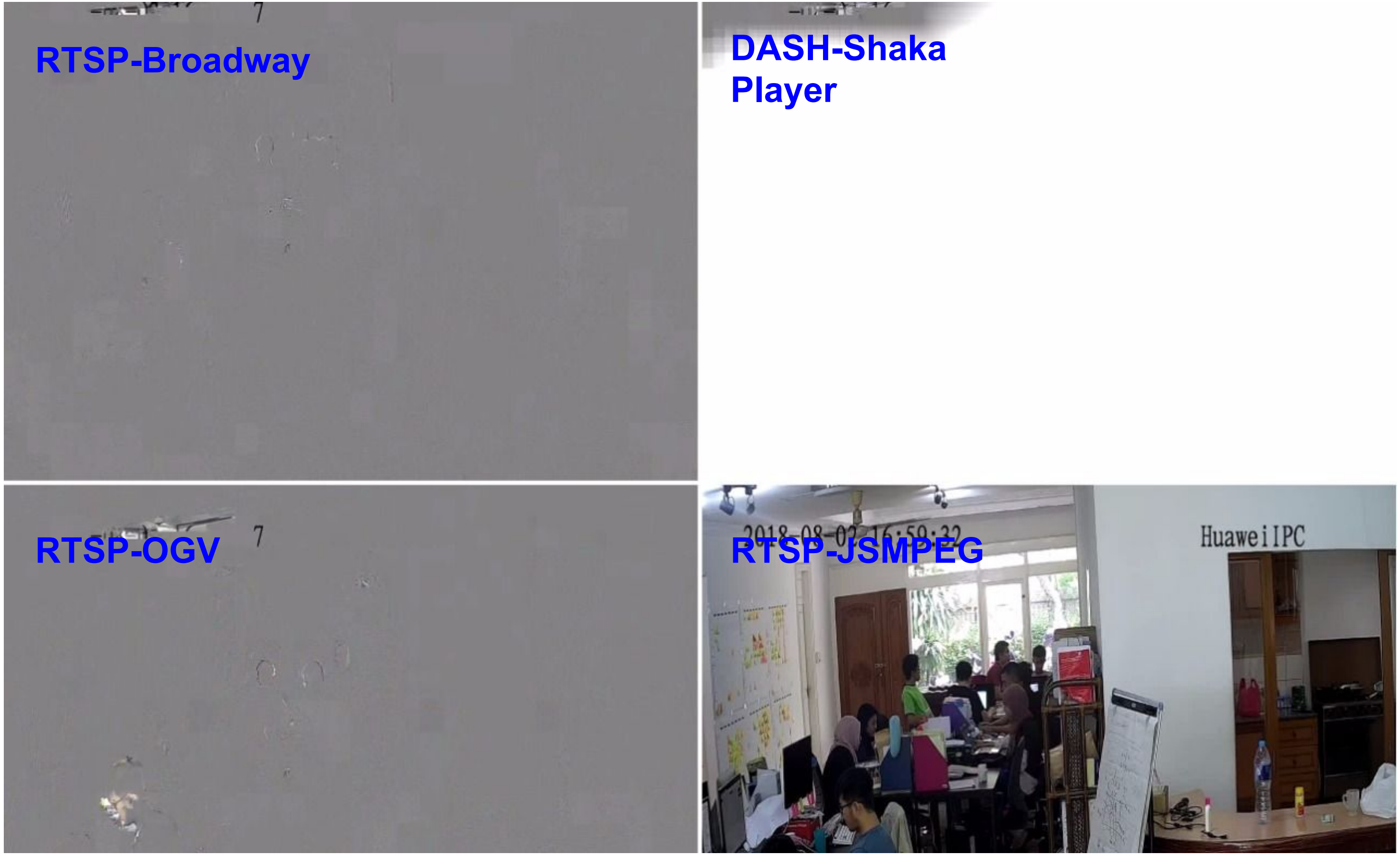
Decoder

- JSMPEG (Codec: MPEG-1)
- Broadway.js (Codec: H264)
- OGV.js (Codec: OGG Vorbis)
- Shaka Player (Codec: H264)

Pemrograman

- Python dan Javascript

# Hasil Penelitian



# Hasil Penelitian

| Decoder | JSMPEG | BROADWAY | OGV | SHAKA | HTML5 |
|---|---|---|---|---|---|
| Protokol | RTSP | RTSP | RTSP | DASH | WebRTC |
| Codec | MPEG-1 | H264 | OGG Vorbis | H264 | - |
| Realtime | Ya | Tidak | Tidak | Tidak | Ya |
| Analisis Lainnya | Menggunakan WebSocket Relay, Kompatibilitas browser baik | H264 decoder dengan Javascript Resource heavy di client. | OGG Vorbis decoder di Javascript, Menggunakan WebAssembly | Adaptive bitrate, Video dikirim dengan referensi dari manifest.mpd | Didesain untuk client-to-client library aiortc. menggunakan negotiation. |

# Proyek Penelitian: Inferencing

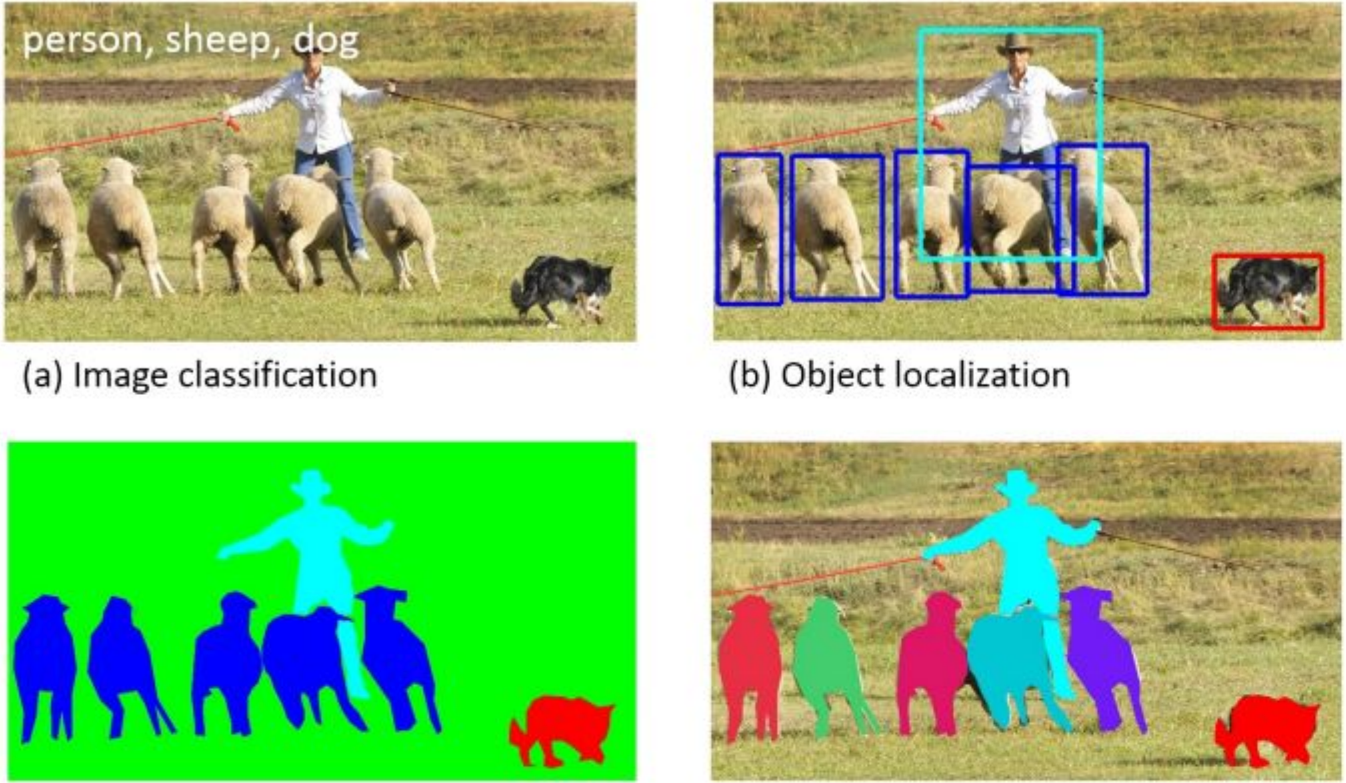
Inferensia Object Detection

- Input Video 1280x720 piksel
- Dataset COCO (Common Object In Context)
  (80 Objek, 5 Captions/image)
- Model CNN Pretrained (MobileNet v2)
- Ouput Model Koordinat Bounding Box
- Pemrograman Python dengan TensorFlow
- Inferensia dilakukan pada server
  (AWS t3.large 2 vCPU, 8 GiB RAM)
- Transmisi frame dilakukan dengan sistem message queue (RabbitMQ dan Redis)

MobileNet

- Arsitektur Convolutional Neural Network
- Didesain untuk mobile phone (Google Pixel) sehingga cocok untuk resource komputasi yang minim.
- Memiliki akurasi yang relatif tinggi apabila dibandingkan dengan Multiply Add Computation (MAC) yang rendah.
- Filter konvolusi dibagi menjadi depthwise dan pointwise.
- Blok Layer menggunakan Batchnorm (BN) dan Activation Function Rectified Linear Unit (ReLU)

# Proyek Penelitian: Inferencing



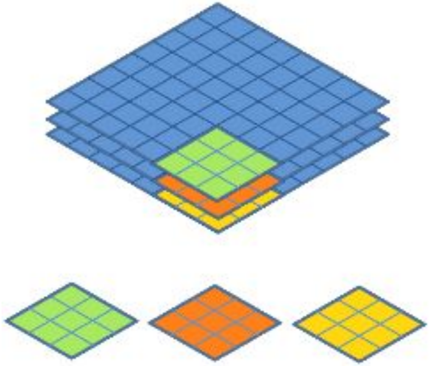

(a) Image classification

(b) Object localization

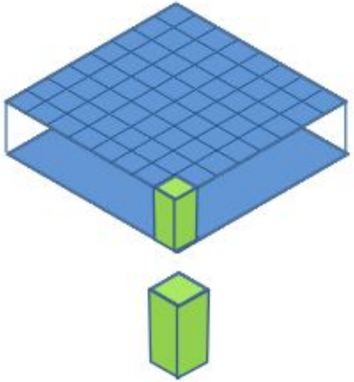
(c) Semantic segmentation

(d) This work

# Proyek Penelitian: Inferencing



Depthwise Convolutional Filters



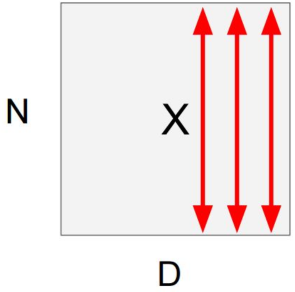
Pointwise Convolutional Filters


Depthwise Separable Convolution

Table 1. MobileNet Body Architecture

| Type / Stride | Filter Shape | Input Size |
|---|---|---|
| Conv / s2 | $3 \times 3 \times 3 \times 32$ | $224 \times 224 \times 3$ |
| Conv dw / s1 | $3 \times 3 \times 32$ dw | $112 \times 112 \times 32$ |
| Conv / s1 | $1 \times 1 \times 32 \times 64$ | $112 \times 112 \times 32$ |
| Conv dw / s2 | $3 \times 3 \times 64$ dw | $112 \times 112 \times 64$ |
| Conv / s1 | $1 \times 1 \times 64 \times 128$ | $56 \times 56 \times 64$ |
| Conv dw / s1 | $3 \times 3 \times 128$ dw | $56 \times 56 \times 128$ |
| Conv / s1 | $1 \times 1 \times 128 \times 128$ | $56 \times 56 \times 128$ |
| Conv dw / s2 | $3 \times 3 \times 128$ dw | $56 \times 56 \times 128$ |
| Conv / s1 | $1 \times 1 \times 128 \times 256$ | $28 \times 28 \times 128$ |
| Conv dw / s1 | $3 \times 3 \times 256$ dw | $28 \times 28 \times 256$ |
| Conv / s1 | $1 \times 1 \times 256 \times 256$ | $28 \times 28 \times 256$ |
| Conv dw / s2 | $3 \times 3 \times 256$ dw | $28 \times 28 \times 256$ |
| Conv / s1 | $1 \times 1 \times 256 \times 512$ | $14 \times 14 \times 256$ |
| $5\times$ Conv dw / s1 | $3 \times 3 \times 512$ dw | $14 \times 14 \times 512$ |
| $5\times$ Conv / s1 | $1 \times 1 \times 512 \times 512$ | $14 \times 14 \times 512$ |
| Conv dw / s2 | $3 \times 3 \times 512$ dw | $14 \times 14 \times 512$ |
| Conv / s1 | $1 \times 1 \times 512 \times 1024$ | $7 \times 7 \times 512$ |
| Conv dw / s2 | $3 \times 3 \times 1024$ dw | $7 \times 7 \times 1024$ |
| Conv / s1 | $1 \times 1 \times 1024 \times 1024$ | $7 \times 7 \times 1024$ |
| Avg Pool / s1 | Pool $7 \times 7$ | $7 \times 7 \times 1024$ |
| FC / s1 | $1024 \times 1000$ | $1 \times 1 \times 1024$ |
| Softmax / s1 | Classifier | $1 \times 1 \times 1000$ |

# Proyek Penelitian: Inferencing




1. compute the empirical mean and variance independently for each dimension.

2. Normalize

$$\widehat{x}^{(k)} = \frac{x^{(k)} - \mathrm{E}[x^{(k)}]}{\sqrt{\mathrm{Var}[x^{(k)}]}}$$

**Input:** Values of $x$ over a mini-batch: $\mathcal{B} = \{x_{1\ldots m}\}$;
Parameters to be learned: $\gamma$, $\beta$

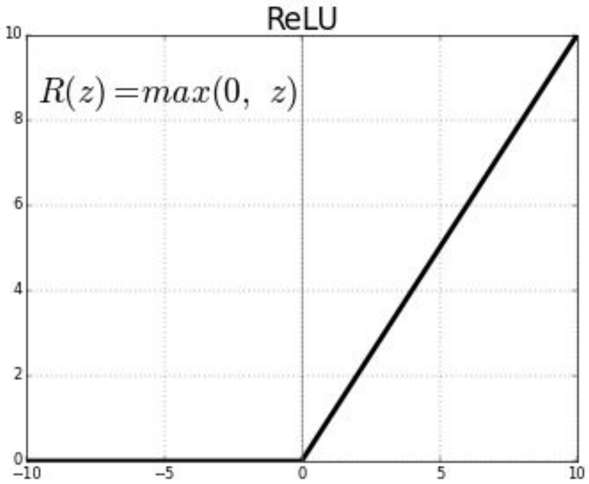
**Output:** $\{y_i = \mathrm{BN}_{\gamma,\beta}(x_i)\}$

$$\mu_{\mathcal{B}} \leftarrow \frac{1}{m}\sum_{i=1}^{m} x_i \qquad \text{// mini-batch mean}$$

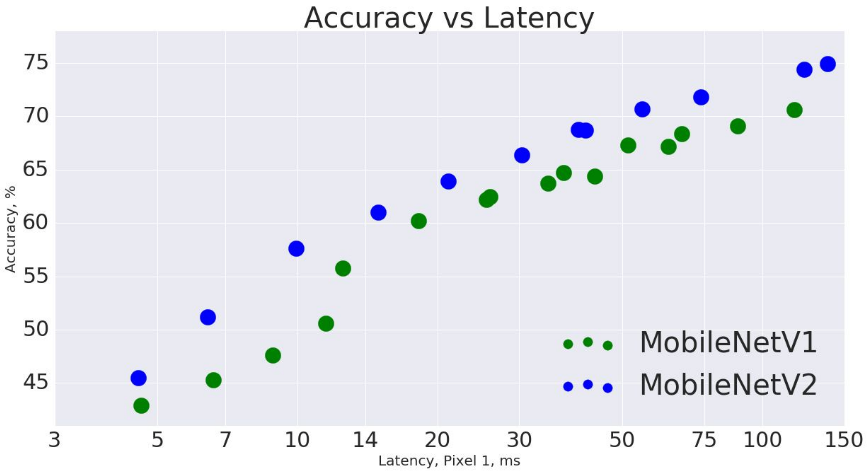
$$\sigma_{\mathcal{B}}^2 \leftarrow \frac{1}{m}\sum_{i=1}^{m} (x_i - \mu_{\mathcal{B}})^2 \qquad \text{// mini-batch variance}$$

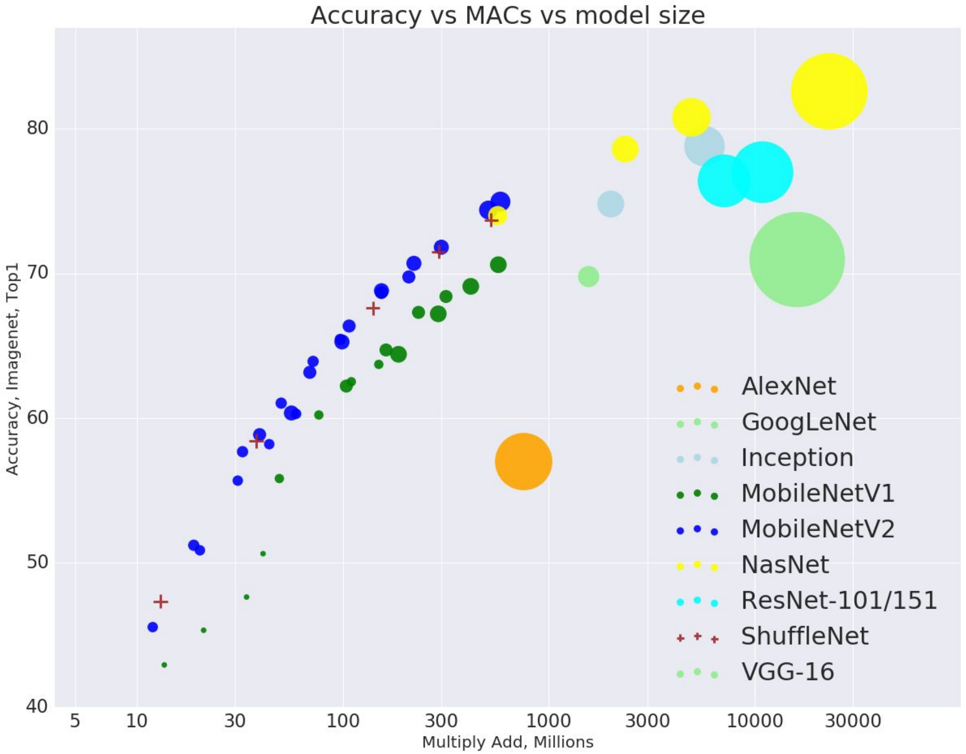
$$\widehat{x}_i \leftarrow \frac{x_i - \mu_{\mathcal{B}}}{\sqrt{\sigma_{\mathcal{B}}^2 + \epsilon}} \qquad \text{// normalize}$$

$$y_i \leftarrow \gamma \widehat{x}_i + \beta \equiv \mathrm{BN}_{\gamma,\beta}(x_i) \qquad \text{// scale and shift}$$

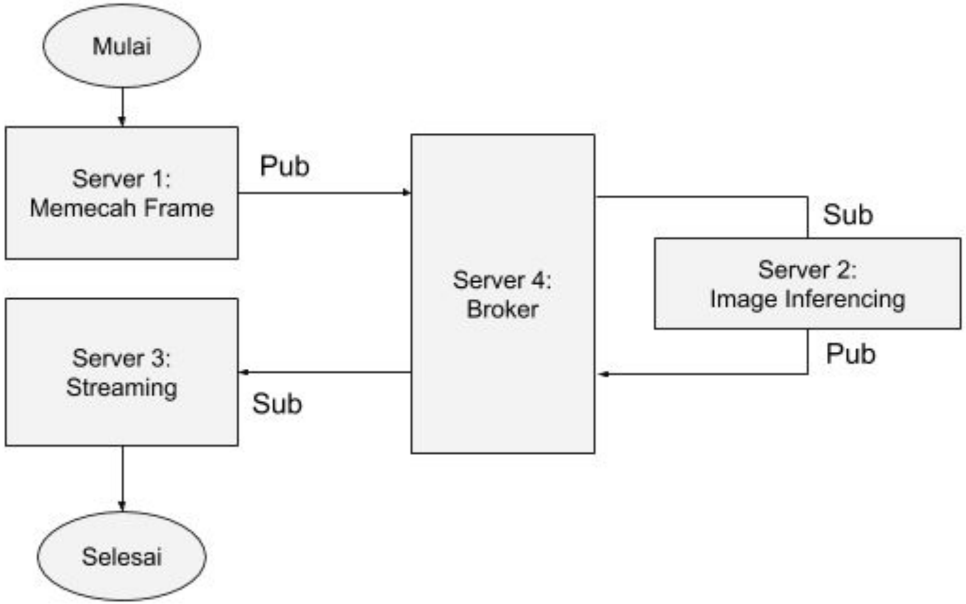


# Proyek Penelitian: Inferencing

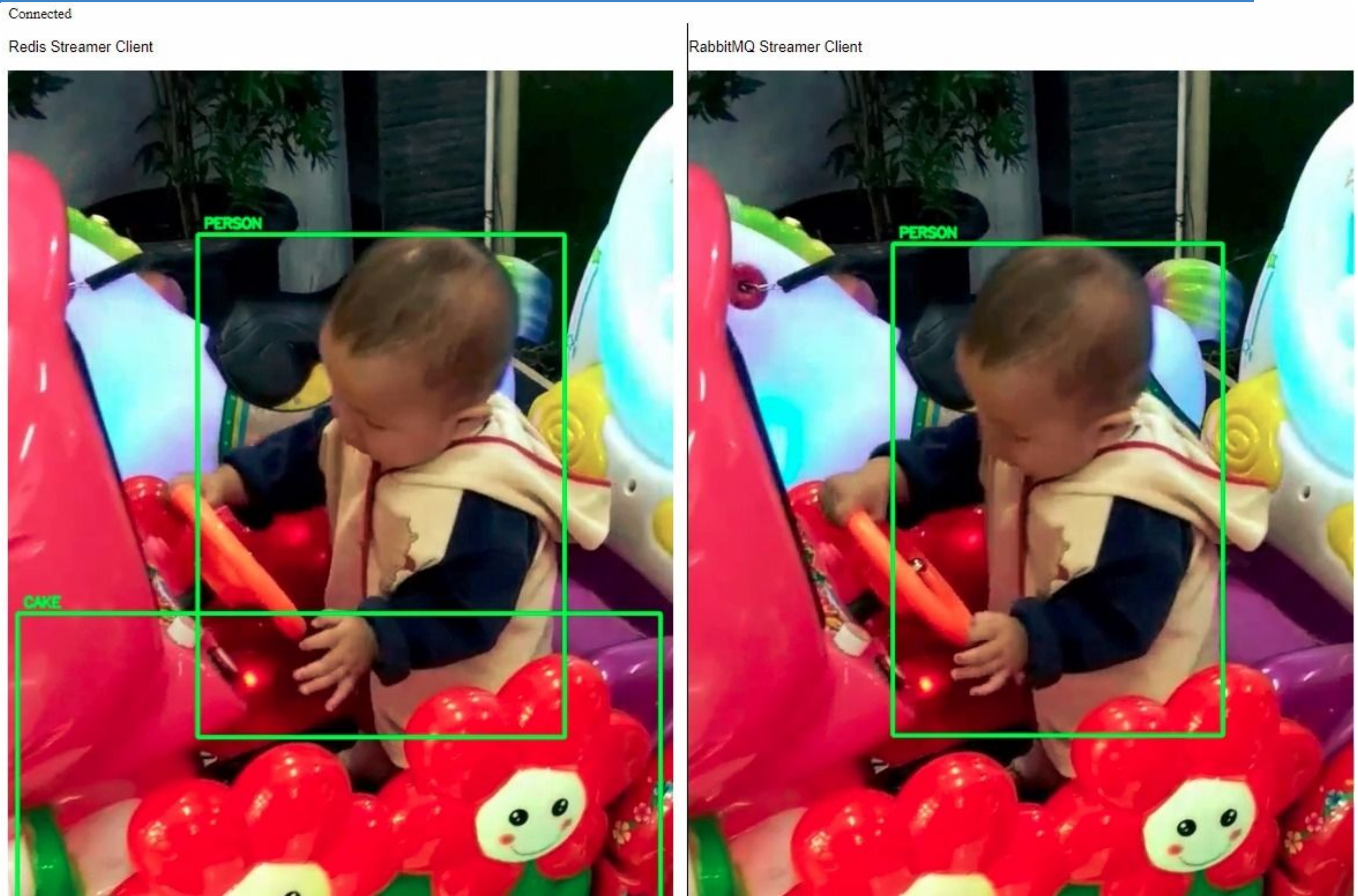


# Hasil Penelitian

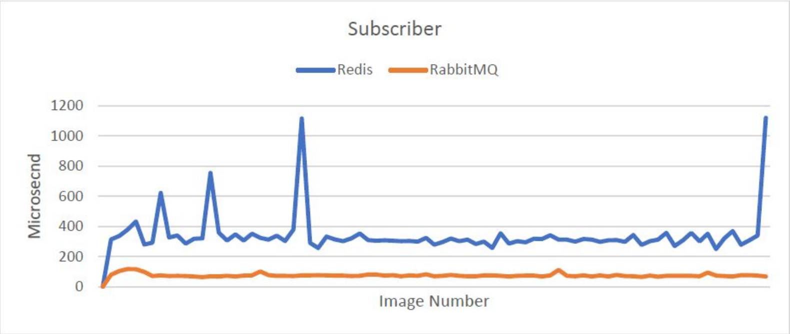


# Hasil Penelitian



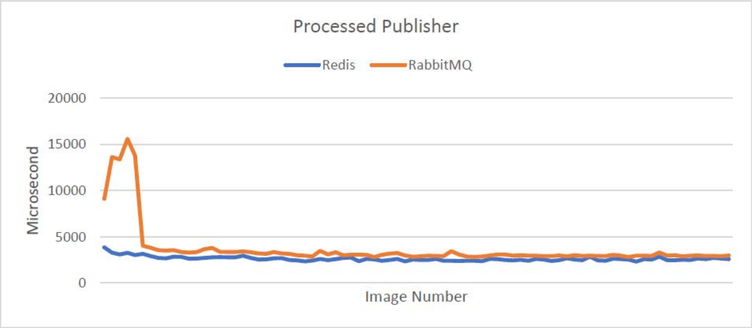
# Hasil Penelitian

Subscriber



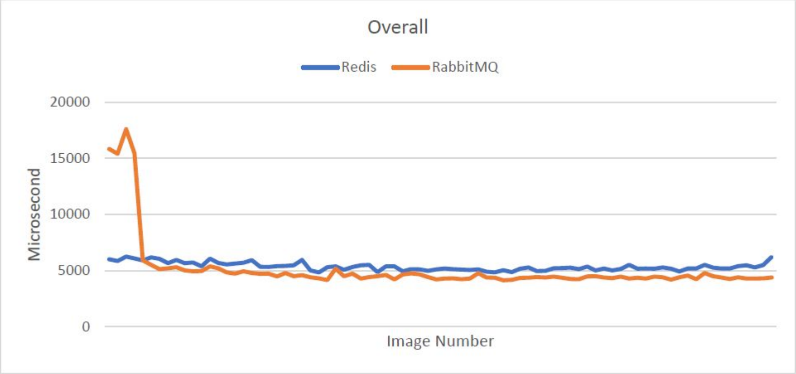

Average:
- Redis : 339,78 µs (+265,80 µs)
- RabbitMQ : 73,98 µs

Processed Publisher




Average:
- Redis : 2.630,61 µs
- RabbitMQ : 3.726.87 µs (+1.096,26 µs)

Overall




Average:
- Redis : 5.371,44 µs (+818,68 µs)
- RabbitMQ : 4.552,76 µs

Akurasi: 75%
Parameter: 6,06 Juta
Waktu Rata-rata/frame:
- 5,371ms (Redis)
- 4,552ms (RabbitMQ)

Waktu Tertunda (Delay):
- 161,31ms (Redis)
- 136,56ms (RabbitMQ)

# Referensi


1. Microsoft COCO: Common Objects in Context (https://arxiv.org/pdf/1405.0312.pdf)
2. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications (https://arxiv.org/pdf/1704.04861.pdf)
3. MobileNetV2: Inverted Residuals and Linear Bottlenecks (https://arxiv.org/pdf/1801.04381.pdf)
4. A Survey of Distributed Message Broker Queues (https://arxiv.org/pdf/1704.00411.pdf)
5. MobileNet: Deteksi Objek pada Platform Mobile (https://medium.com/nodeflux/mobilenet-deteksi-objek-pada-platform-mobile-bbbf3806e4b3)
6. Convolutional Neural Networks (http://cs231n.github.io/convolutional-networks/)

www.nodeflux.io