ESTIMASI POSE TIGA DIMENSI DARI GAMBAR MONOKULER MENGGUNAKAN DEEP NEURAL NETWORK

[1]Denilson
2Dr. Dharmayanti, ST., MMSI.

[1]JL. TK Al Kindi No. 126 Rt 004/001 Kel. Cipayung Jaya Kec. Cipayung Depok (denilson020898@gmail.com)
[2]Jl. Samiaji VIII/336 Rt 007/019 Kel. Sukmajaya Kec. Mekar Jaya Depok II Tengah (dharmayanti77@gmail.com)
Jurusan Teknik Informatika, Fakultas Teknologi Industri,
Universitas Gunadarma

ABSTRAKSI

Perkembangan teknologi digital yang pesat baik pada aplikasi atau ilmu pengetahuan dapat manghasilkan rekam jejak digital yang bermanfaat. Jumlah data digital yang tersedia sangat banyak dan diprediksi akan semakin bertambah. Salah satu penggunaan data adalah membuat suatu fungsi pemetaan yang mencari korelasi antara suatu domain ke domain lainnya dengan menggunakan data terkait sebagai acuan dasar. Data digital berbentuk rangkaian gambar atau video merupakan data yang bersifat laten yang berarti data tersebut memiliki informasi semantik yang tersembunyi. Penelitian ini membahas pembuatan sebuah fungsi yang memetakan gambar dua dimensi terhadap titik kunci pose tiga dimensi yang bersifat laten menggunakan permodelan deep neural network. Perangkat lunak yang dibangun dengan pemrosesan data, perancangan arsitektur model, pemelajaran model secara mandiri, dan menampilkan visualisasi penggunaan model. Arsitektur model yang digunakan terdiri dari beberapa blok residual network yang menambahkan input terhadap output masing-masing blok. Hasil dari uji coba menjelaskan bahwa teori dan data yang dipakai benar dan penggunaan aplikasi terhadap data baru sesuai prediksi.
Kata Kunci: Estimasi Pose, Gambar Monokuler, Jaringan Saraf Tiruan,
Pemelajaran Dalam, Visi Komputer.

ABSTRACT

Digital technologies have been developed rapidly in application and science may produce digital track records that are actually useful. Digital data are available in a huge number and are predicted to increase. One way to utilize this data is to create a mapping function that finds a correlation between domains from the data itself as a reference. Digital data in form of sequence of images or videos are latent which mean data itself has some hidden semantic meanings. This research is about making a mapping function that maps two dimensional images into three dimensional human pose keypoints using deep neural network modeling. The software is built in steps that involve data preprocessing, model architecture design, self-traning deep neural network, and visualization. The model consists of some blocks of residual networks that sum up its inputs and outputs. The result from testing explains that the theories and data are correct and runs correctly using new data as input.
Keywords: Artificial Neural Network, Computer Vision, Deep Learning, Monocular Image, Pose Estimation

PENDAHULUAN

Pemanfaatan teknologi yang terkomputerisasi oleh manusia selalu meninggalkan jejak yang tersimpan dalam bentuk data digital. Rekam jejak ini merupakan bukti perilaku dan karakteristik manusia sehingga dijadikan sebagai acuan pengembangan teknologi dan ilmu pengetahuan pada masa mendatang. Data digital yang umumnya dimanfaatkan oleh manusia meliputi teks, citra audio, citra visual, dan citra audio visual yang disimpan ke dalam suatu media penyimpanan. Banyaknya jumlah data yang tersedia dan diprediksi akan semakin bertambah membuat gaya hidup manusia semakin bergantung pada teknologi digital.

Permodelan pemelajaran dalam ataudeep learningdapat memetakan suatu domain ke domain lainnya secara mandiri menggunakan pemelajaran jaringan saraf tiruan dalam atau deep neural network. Pemelajaran dalam dapat dilakukan dengan komputasi mandiri yang sangat bergantung pada kuantitas dan kualitas data yang baik. Pemelajaran dalam menggunakan jaringan saraf tiruan dapat digunakan untuk mengembangkan teknologi khususnya di bidang visi komputer seperti melakukan estimasi pose tiga dimensi tubuh manusia yang terdapat dalam suatu gambar monokuler.

METODE PENELITIAN

Penelitian ini membahas pemanfaatan data gambar sebagai acuan dalam melakukan pemelajaran dan implementasi model deep neural network untuk mencari dan memetakan koordinat tiga dimensi pose tubuh manusia dalam sebuah rangkaian gambar secara lokal. Pose yang digunakan tidak bersifat grounded yang berarti koordinat pose tidak berpusat pada titik lantai tertentu. Pengerjaan aplikasi mengutamakan dua langkah penting yang meliputi pengolahan data dan pembuatan model. Aplikasi yang dibuat dapat menampilkan plot grafik tiga dimensi menyerupai struktur anatomi tubuh manusia sesuai dengan pose hasil estimasi dari gambar masukkan. Hasil pemelajaran model ditampilkan dalam grafik dua dimensi untuk analisis lebih lanjut.

Pemelajaran model deep neural network diimplementasikan menggunakan framework PyTorch. Kedua dataset yang digunakan diolah terlebih dahulu sehingga memenuhi syarat PyTorch dalam melakukan deep learning. Setiap model kemudian digunakan terhadap dataset inferensi aplikasi. Proses dan hasil estimasi diurai lebih lanjut dalam bentuk grafik visual.

PEMBAHASAN

Analisis Data

Data pembuatan model yang digunakan adalah data pemetaan dari pose dua dimensi ke pose tiga dimensi. Pose dua dimensi merupakan sampel, sedangkan pose tiga dimensi merupakan target. Sumber data adalah Human3.6M Dataset mengenai informasi perakaman gerakan pose manusia yang menyimpan gambar dari beberapa sisi beserta dengan pose dua dimensi dan tiga dimensinya (Ionescu et al, 2014).

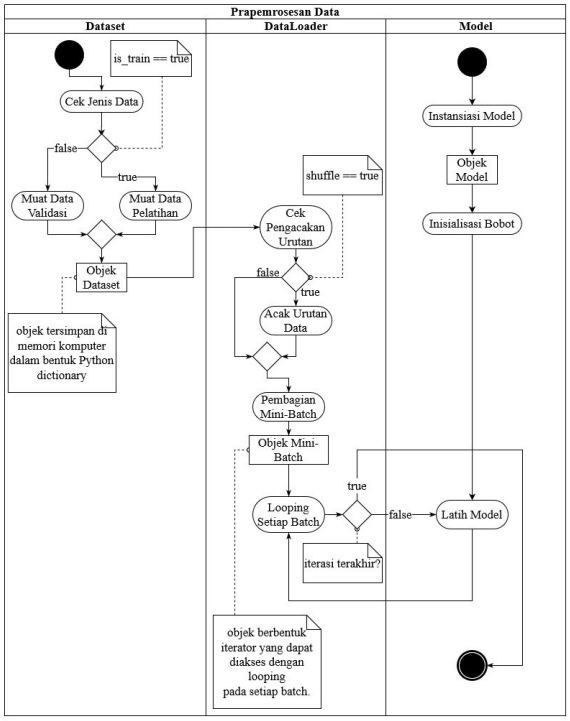
Tabel 1. Data Pemelajaran Model

Nama Berkas	Isi
rcams.pt	Matriks kamera posisi
stat_2d.pt	Mean, std, dan pose 2D
stat_3d.pt	Mean, std, dan pose 3D
test_2d.pt	Data pose 2D untuk test

test_3d.pt	Data pose 3D untuk test
train_2d.pt	Data pose 2D untuk train
train_3d.pt	Data pose 3D untuk train

Pra Pemrosesan Data

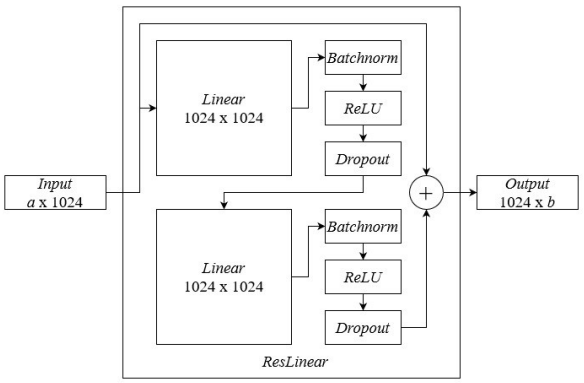
Kelas Dataset dan DataLoader pada framework PyTorch memiliki fungsionalitas untuk melakukan pembacaan dan pembagian rangkaian data secara stochastic. Kelas DataLoader dapat memindahkan informasi didalam kelas Dataset ke VRAM pada GPU berbentuk mini-batch sehingga dapat diproses secara stochastic dan paralel. Langkah pertama yang dilakukan adalah melakukan cek apakah data yang diinginkan adalah data pelatihan atau data validasi. DataLoader menerima objek tersebut kemudian melakukan pengacakan data dan pembagian mini-batch. Hasil objek dapat diiterasi untuk melakukan pelatihan model (Kingma et al. 2014).



Gambar 1. Pra Pemrosesan Data

Arsitektur Model

Arsitektur model jaringan saraf tiruan yang digunakan memiliki input berbentuk vektor dengan ukuran tiga puluh dua dan output berbentuk vektor dengan ukuran empat puluh delapan. Rangkaian lapisan yang menghubungkan input dan output berupa lapisan residual network. Bobot setiap lapisan diinisialisasi secara acak dengan distribusi normal. Sebuah lapisan residual network merupakan jaringan dengan arsitektur kecil dan sederhana yang dapat dipasang atau dibongkar secara modular yang disebut ResLinear. Lapisan ResLinear melakukan operasi penjumlahan antara input dan output. Sebuah ResLinear memiliki dua lapisan linier, dua lapisan Batchnorm, dua lapisan Dropout, dan dua lapisan ReLU. Komponen-komponen penyusun sebuah lapisan ResLinear dengan ukuran input a dan ukuran output b (Martines et al. 2017).

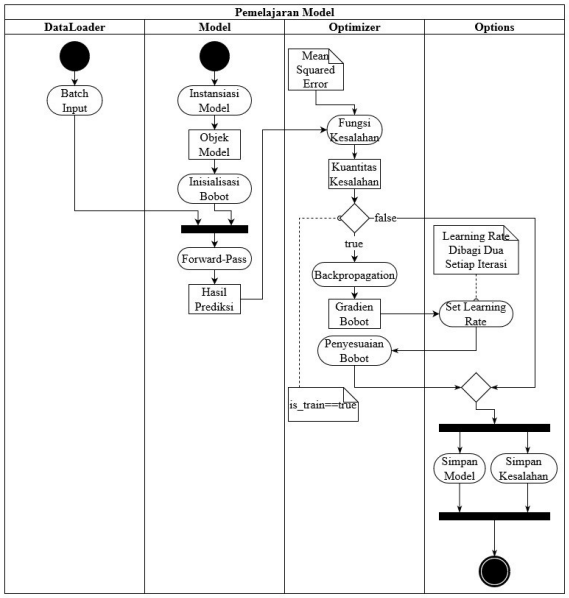


Gambar 2. Arsitektur Model

Pemelajaran Model

Pemelajaran model merupakan kelanjutan dari pra pemrosesan data. Sebuah model baru diinstansiasi sehingga menghasilkan objek model. Objek model tersebut kemudian menginisialisasi bobot parameter dengan bilangan acak dari distribusi normal. Batch input yang berasal dari interasi DataLoader diproses oleh model dengan metode feed-forward. Hasil prediksi sementara dari model

didapatkan yang kemudian dibandingkan tingkat kebenarannya menggunakan fungsi kesalahan. Fungsi kesalahan mean squared error menghasilkan angka kuantitas kesalahan. Angka ini merupakan tolak ukur seberapa akurat kemampuan model dalam menghasilkan output yang relevan. Apabila model tidak berada dalam status "is_train", maka kuantitas kesalahan yang didapatkan langsung disimpan dalam bentuk array untuk keperluan analisis. Apabila model berada dalam status "is_train", maka model melakukan backpropagation untuk menghasilkan gradien bobot. Learning rate kemudian dibagi dengan dua sehingga menjadi lebih kecil. Penyesuaian bobot dilakukan dengan menjumlahkan bobot dengan hasil operasi perkalian antara learning rate dan gradien bobot. Bobot model yang telah diperbarui disimpan berserta dengan kuantitas kesalahannya. Algoritma yang sama akan diulangi pada setiap batch input.



Gambar 3. Skema Pemelajaran Model

Pra Pemrosesan Data Inferensi

Proses inferensi pose dua dimensi menjadi lebih cepat ketika gambar dua dimensi yang dioperasikan memiliki resolusi yang kecil. Tingkat resolusi yang digunakan saat merekam pose adalah 640 x 360. Memperkecil ukuran resolusi juga harus dilihat dari segi kualitas gambar yang dihasilkan. Gambar juga memiliki banyak informasi statis seperti pada area piksel berwarna hitam yang berada di kiri dan kanan gambar. Resolusi 290 x 290 dengan titik tengah berada pada titik kunci pinggang dianggap tepat karena mampu menjangkau semua pose badan dan kualitas gambar yang masih bagus. Inferensi yang lebih efisien dapat dilakukan dengan memperkecil resolusi dan area piksel mati.

Gambar 4. Pemrosesan Data Inferensi

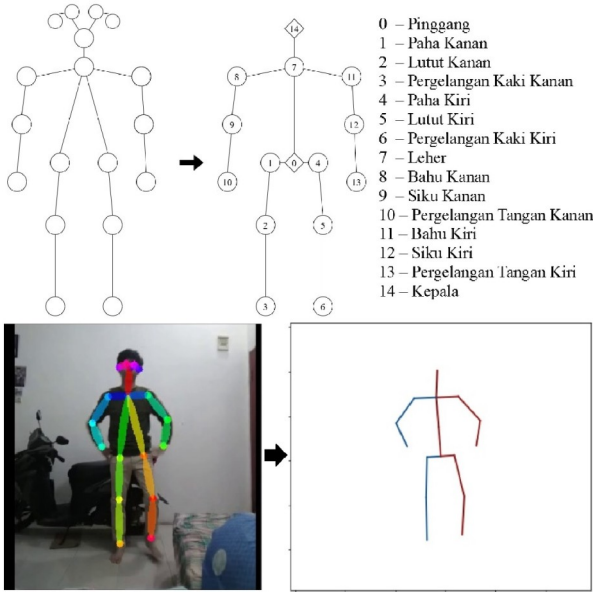
OpenPose

OpenPose merupakan aplikasi estimasi pose tubuh manusia dua dimensi. OpenPose menerima input gambar kemudian mencari titik kunci pose dua dimensi. Titik kunci pose dua dimensi berada pada koordinat lokal sesuai dengan bidang gambar. OpenPose menghasilkan berkas "json" yang berisi hierarki pose menurut spesifikasi COCO-MS. Anotasi COCO-MS berisi delapan belas titik kunci tubuh manusia dengan urutan tertentu (Cao et al. 2019).

Gambar 5. Spesifikasi Titik Kunci COCO-MS

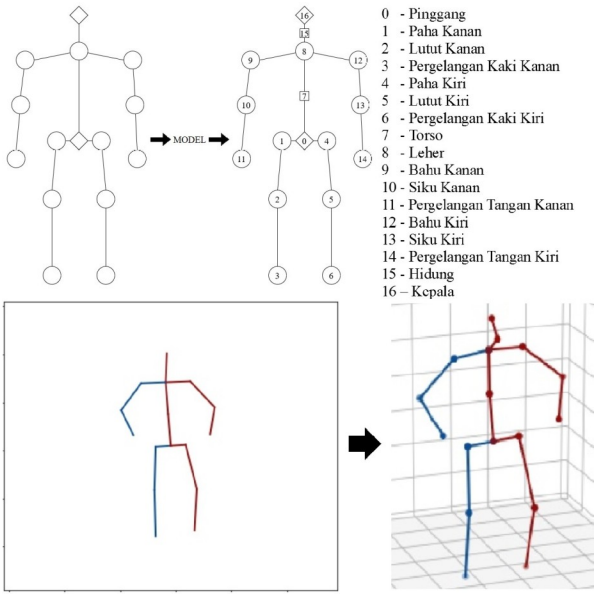
Inferensi Model

Rangkaian titik kunci yang dihasilkan oleh OpenPose memiliki spesifikasi COCO-MS yang berbeda dengan spesifikasi dataset pembuatan model. Spesifikasi COCO-MS yang memiliki delapan belas titik kunci dikonversi menjadi lima belastitik kunci dengan menyatukan titik kunci kedua mata dan kedua telinga serta membuat titik kunci pinggang berdasarkan rata-rata kedua kaki bagianatas. Konversi ini dilakukan supaya model dapat melakukan inferensi terhadap titik kunci yang mewakili pose tersebut. Visualisasi konversi titik kunci COCO-MS menggunakan warna biru mewakili anggota badan sebelah kanan bersamaan dengan warna merah yang mewakili anggota badan tengah dan kiri.



Gambar 6. Konversi Titik Kunci 2D

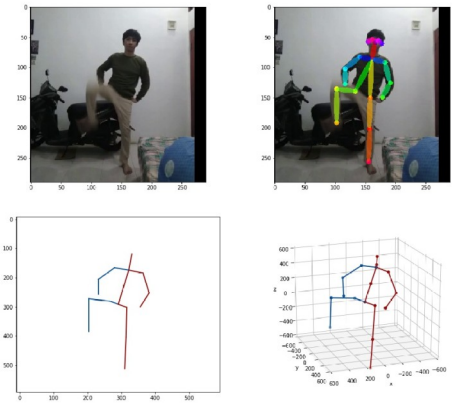
Inferensi pada model deep neural network menerima titik kunci pose duadimensi yang telah dikonversi sebagaiinputkemudian menghasilkan titik kuncipose tiga dimensi sebagaioutput. Inferensi model menghasilkan tujuh belas titikyang berada dalam bidang tiga dimensi. Titik kunci pose tiga dimensi memiliki duatitik baru yang meliputi torso dan hidung. Titik torso mewakili lengkungan badanpada pose tiga dimensi sehingga pose terlihat akurat. Titik hidung mewakili arahhadapan kepala. Kedua titik tambahan ini memperjelas orientasi pose tiga dimensi.



Gambar 7. Inferensi Model

Visualisasi

Visualisasi mencakup penggunaan aplikasi dalam satu bingkai secara keseluruhan.Terdapat empat buah figur yang masing-masing mewakili langkah-langkah pada tahapan uji coba. Figur pertama (kiri atas) berisi prapemrosesan data inferensi dimana resolusi gambar diubah menjadi 290 x 290. Figur kedua (kanan atas) menggambarkan pose dua dimensi yang didapatkan oleh OpenPose. Figur ketiga (kiri bawah) menggambarkan konversi titik kunci OpenPose dengan spesifikasi COCO-MS menjadi titik kunci yang cocok dengan model. Figur keempat (kanan bawah) menggambarkan pose tiga dimensi yang dihasilkan oleh model kedalam sebuah sistem koordinat tiga dimensi.



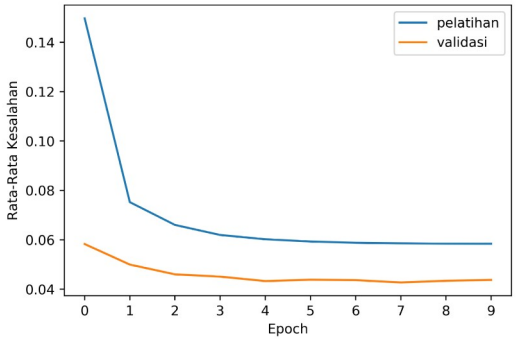
Gambar 8. Visualisasi Aplikasi

Script Python

Langkah-langkah yang dilakukan pada tahapan produksi masih terpisah dalam beberapa cell di interactive development environment Jupyter Lab. Cell tersebut kemudian digabungkan kedalam sebuah script yang bernama "run.py". Hal ini dilakukan agar pengguna aplikasi dapat menjalankan aplikasi dengan mudah. Aplikasi dapat dijalankan dengan memasukkan perintah "python run.py" dengan syarat semua dependencies python versi 3 telah terpasang.

Hasil Pemelajaran Model

Pemelajaran model yang dilakukan selama sepuluh epochs menunjukkan bahwa kesalahan model dalam mengestimasi pose tiga dimensi berkurang dalam setiap epoch. Learning rate yang dibagi dua dalam setiap epoch mempengaruhi pemelajaran model dimana penyesuaian model semakin teliti. Adaptasi bobot model terjadi secara drastis padaepoch0 sampai dengan epoch3. Epoch4 dan seterusnya menggunakan learning rate yang semakin kecil sehingga model semakin teliti dalam melakukan adaptasi.

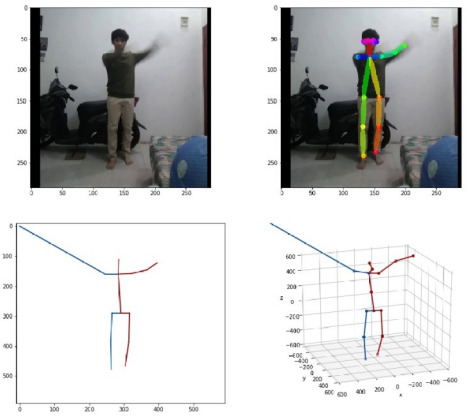


Gambar 9. Grafik Pemelajaran Model

Analisis Uji Coba Aplikasi

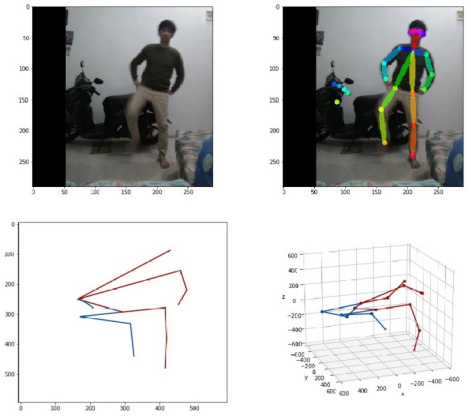
Inferensi yang bagus akan terjadi apabila langkah-langkah pada tahapan ujicoba tidak mengalami kesalahan. Kualitas gambar dan pose yang tidak cacat juga mempengaruhi proses dari awal hingga akhir. Prapemrosesan gambar pada data inferensi yang tepat memudahkan OpenPose dalam mencari titik kunci pose dua dimensi secara lengkap. Titik kunci OpenPose yang lengkap kemudian memenuhi syarat untuk dikonversi menjadi spesifikasi yang diinginkan. Informasi tersebut kemudian diteruskan ke model untuk mendapatkan titik kunci pose tiga dimensi.

Kualitas pose yang cacat menghasilkan estimasi pose tiga dimensi yang cacat. Oklusi pose pada gambar dua dimensi dapat menghilangkan suatu bagiantubuh. Hilangnya bagian ini dari gambar menyebabkan kesalahan pada langkah-langkah selanjutnya. Titik kunci lengan kanan hilang ketika pose lengan mengarah lurus ke lensa kamera sehingga terjadi oklusi. Hal ini menyebabkan OpenPose tidak dapat menemukan titik kunci lengan kanan dan memberi nilai nol pada titik kunci tersebut. Proses konversi dan inferensi titik kunci tiga dimensi juga menghasilkan pose yang tidak realistis.



Gambar 10. Inferensi Pose Hilang

Kesalahan juga dapat terjadi pada proses inferensi titik kunci. Apabila OpenPose mengeluarkan output yang ambigu dimana terdapat titik kunci yang dianggap sebagai bagian dari tubuh manusia. OpenPose menghasilkan titik kunci ganda yang tidak sesuai dengan spesifikasi yang diperlukan meskipun berhasil mendeteksi pose secara lengkap. Hasil keluaran yang tidak sesuai dengan spesifikasi model mengakibatkan estimasi pose tiga dimensi yang rusak.



Gambar 11. Kesalahan Deteksi

PENUTUP

Aplikasi estimasi pose tiga dimensi menggunakan model deep neural network berhasil dilatih.Model ini melakukan pemelajaran secara mandiri menggunakan informasi pose dua dimensi sebagai input dan pose tiga dimensi sebagai output. Model tersebut diuji untuk menemukan pose tiga dimensi pada data inferensi. Model melakukan pemelajaran selama sepuluh epochs dengan hasil rata-rata kesalahan akhir bernilai 0.0584 pada data pelatihan dan 0.0437 pada data validasi. Nilai kesalahan data pelatihan masih lebih besar daripada nilai data validasi. Hal ini menandakan model masih berada pada kondisi underfitting dimana selisih kedua nilai tersebut relatif besar. Model yang lebih baik dapat didapatkan dengan melatih model dalam jumlah epochs yang lebih banyak dan berhenti saat mulai terjadi overfitting.

Pengembangan model deep neural network ini masih menggunakan arsitektur minimalis, data dengan satu domain, dan memiliki tahapan yang tidak efisien. Pengembangan selanjutnya disarankan menggunakan arsitektur yang lebih efisien. Arsitekur residual network merupakan arsitektur yang paling bagus pada saat penulisan ini dilakukan. Algoritma pemelajaran yang lebih efisien juga disarankan pada penelitian mendatang. Data dengan domain yang lebih luas juga merupakan hal yang penting seperti estimasi pose pada hewan tertentu.

DAFTAR PUSTAKA

[1] Cao, Z., Hidalgo Martinez, G., Simon, T., Wei, S., dan Sheikh, Y. A. (2019).Openpose: Realtime multi-person 2d pose estimation using part affinity fields.IEEE Transactions on Pattern Analysis and Machine Intelligence.

[2] Ionescu, C., Papava, D., Olaru, V., dan Sminchisescu, C. (2014). Human3.6m: Large scale datasets and predictive methods for 3d human sensing in natural environments.IEEE Transactions on Pattern Analysis and Machine Intelligence,36(7):1325–1339.

[3] Kingma, D. P. dan Ba, J. (2014).Adam: A Method for StochasticOptimization.arXiv e-prints, page arXiv:1412.6980.

[4] Martinez, J., Hossain, R., Romero, J., dan Little, J. J. (2017). A simple yeteffective baseline for 3d human pose estimation. In ICCV.