

Kinerja *Deep Convolutional Network* untuk Pengenalan Aksara Pallawa

Wiwien Widyastuti

Program Studi Teknik Elektro, Fakultas Sains dan Teknologi, Universitas Sanata Dharma,
Kampus III Universitas Sanata Dharma, Paingan, Maguwoharjo, Depok, Yogyakarta 55282,
Telp. (0274) 883037, Fax. (0274) 886529
e-mail: wiwien@usd.ac.id

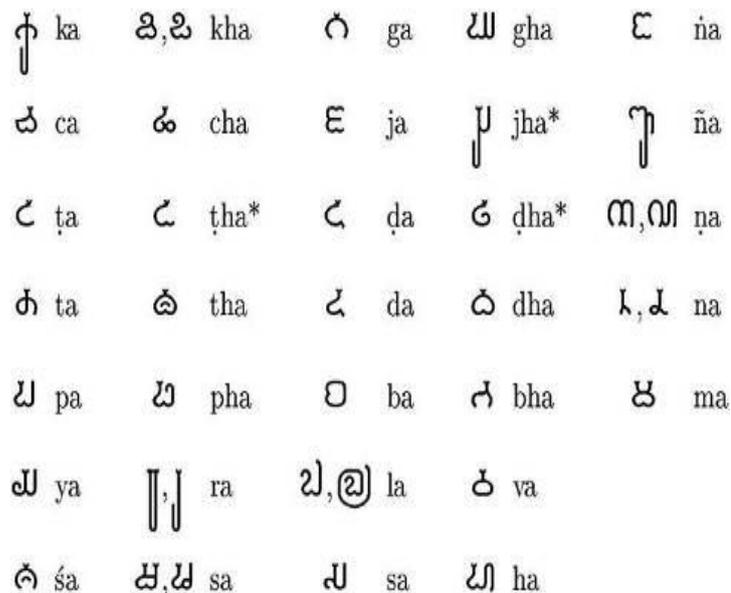
Abstract

This research trained Deep Convolutional Networks(ConvNets) to classify hand-written Pallava alphabet. The Deep ConvNets architecture consists of two convolutional layers, each followed by max-pooling layer, two Fully-Connected layers. It had 442.602 parameters. This model classified 660 images of hand-written Pallava alphabet into 33 diferent classes. To make training faster, this research used GPU implementation with 384 CUDA cores. Two different techniques were implemented, Stochastic Gradient Descent (SGD) and Adaptive Gradient, each trained with 10, 20, 30 and 40 epoch. The best accuracy was 67,5%, achieved by the model with SGD technique trained at 30 epoch.

Keywords: Deep ConvNets, Pallava, GPU, SGD

1. Pendahuluan

Aksara Pallawa atau sering disebut dengan Pallava adalah salah satu jenis aksara yang berasal dari India bagian Selatan. Banyak prasasti di Nusantara yang ditulis menggunakan aksara Pallawa, sehingga aksara ini merupakan warisan budaya sangat penting bagi bangsa Indonesia. Aksara Pallawa merupakan ibu dari hampir semua aksara daerah di Indonesia [1] Sangat disayangkan bahwa tidak banyak yang mengetahui aksara Pallawa tersebut. Aksara Pallawa dapat dilihat pada Gambar 1.



Gambar 1. Aksara Pallawa [1]

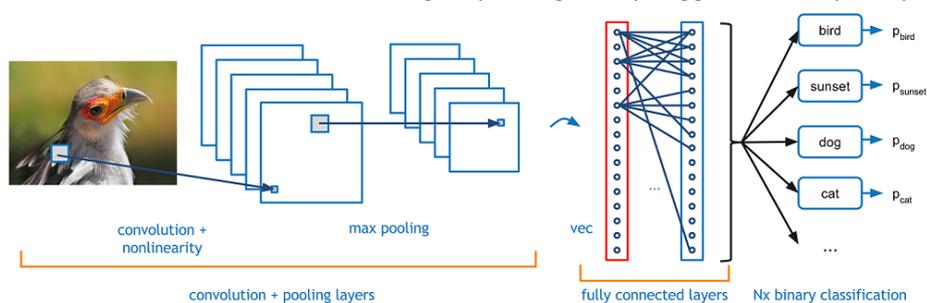
Saat ini perkembangan teknologi komputer yang pesat, menjadikan kemampuan komputasi pada komputer juga meningkat. Jumlah *core* prosesor yang banyak memungkinkan pemrosesan paralel yang mengakibatkan waktu pemrosesan menjadi jauh lebih singkat. Kemampuan tersebut sangat mendukung bidang *Artificial Intelligence* khususnya dalam bidang teknologi *machine learning*. Teknologi *machine learning* memiliki kegunaan besar dalam banyak aspek di kehidupan masyarakat modern. Teknologi *machine learning* digunakan untuk mengidentifikasi objek pada suatu citra, percakapan ke dalam teks, mencocokkan berita, mengunggah produk sesuai minat pengguna, dan memilih hasil yang relevan pada suatu pencarian [2].

Banyak algoritma/model yang digunakan pada teknologi *machine learning* di antaranya adalah Model Hidden Markov, *Neural Network* dan *Deep learning*. Teknik *machine learning* konvensional mempunyai kemampuan terbatas dalam mengolah data natural dalam bentuk mentahnya. Pembelajaran representasi (*representation learning*) merupakan suatu metode yang memungkinkan suatu mesin mengolah data mentah dan secara otomatis menemukan representasi yang diperlukan untuk deteksi atau klasifikasi. *Deep learning* merupakan metode pembelajaran representasi yang memungkinkan model komputasional yang tersusun dari banyak lapisan pengolahan untuk belajar representasi data dengan banyak tingkat abstraksi. Metode ini secara signifikan meningkatkan unjuk kerja pada pengenalan percakapan, pengenalan obyek visual, deteksi obyek dan lain sebagainya [2]. Salah satu metode *deep learning* yang seringkali digunakan untuk pengenalan citra adalah *deep convolutional neural networks*.

Penggunaan metode *deep convolutional networks (ConvNets)* untuk pengenalan atau klasifikasi citra telah banyak dilakukan oleh para peneliti. Alex Krizhevsky, Ilya dan Geoffrey menggunakan *deep ConvNets* untuk klasifikasi 1,2 juta citra ke dalam 1000 kelas dengan *error rate* 37,5% [3]. Pierre, Soumith dan Yann LeCun menggunakan *deep ConvNets* untuk klasifikasi digit pada nomor rumah dan memperoleh akurasi 94,85% [4]. Michael Nielsen menggunakan metode ini untuk mengenali angka tulisan tangan dan memperoleh akurasi klasifikasi 99,06% [5]. Semua penelitian yang telah dilakukan menunjukkan bahwa metode *deep ConvNets* memberikan hasil yang baik khususnya pada bidang pengenalan/klasifikasi pola.

Penelitian pengenalan aksara Pallawa dengan Model Hidden Markov juga telah dilakukan, tetapi masih mendapatkan hasil yang kurang baik yaitu dengan akurasi klasifikasi 35,15% [6]. Berdasarkan hal tersebut dan hasil penelitian tentang *deep ConvNets* maka penelitian ini akan menerapkan *deep ConvNets* untuk pengenalan aksara Pallawa.

ConvNets dirancang untuk mengolah data yang berbentuk *multiple array*, sebagai contoh citra berwarna yang tersusun dari 3 buah array 2 dimensi yang berisi intensitas piksel dalam 3 kanal warna (contoh : RGB). Terdapat 4 ide pokok yang melatarbelakangi *ConvNets*: koneksi lokal (*local connection*), *shared weight*, *pooling*, dan penggunaan banyak lapisan.

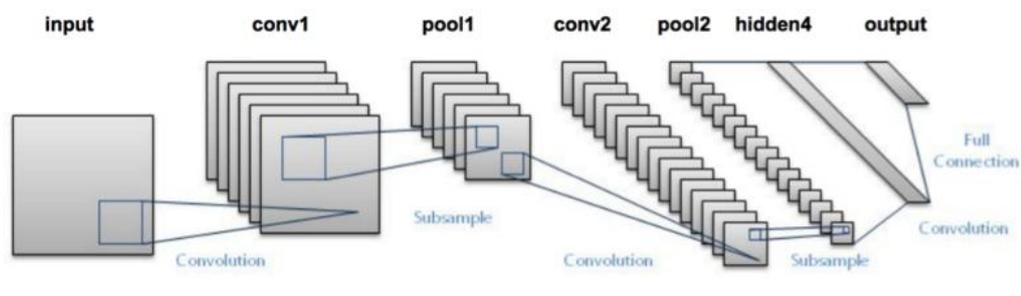


Gambar 2. Contoh Arsitektur *Convolutional Networks* [7]

Contoh arsitektur *ConvNets* dengan sebuah *stage* dan sebuah lapisan *fully-connected* dapat dilihat pada Gambar 2. Sebuah *stage* terdiri dari dua tipe lapisan yaitu lapisan konvolusional dan lapisan *pooling*. Unit pada lapisan konvolusional diorganisasikan dalam

feature map, yang di dalamnya tiap unit dihubungkan dengan suatu area penerimaan lokal (*local receptive field*) dalam *feature map* pada lapisan sebelumnya dengan menggunakan *weight set* yang disebut *kernel* atau *filter* [2], [5]. Lapisan selanjutnya adalah lapisan *pooling*. Lapisan *pooling* digunakan untuk menyederhanakan informasi pada output lapisan konvolusional. Salah satu prosedur untuk *pooling* yang biasa digunakan adalah *max-pooling*. Pada *max-pooling*, unit *pooling* secara sederhana akan berisi aktivasi maksimum dari suatu wilayah input. Karena lapisan konvolusional biasanya terdiri lebih dari sebuah *feature map*, maka *max-pooling* juga diaplikasikan untuk masing-masing *feature map*. Lapisan terakhir adalah lapisan yang *fully-connected*. Lapisan ini menghubungkan setiap neuron dari lapisan *pooling* ke setiap neuron output atau sejumlah kelas klasifikasi [2], [5].

Deep convolutional network merupakan pengembangan dari *ConvNets*. Jika selanjutnya lapisan konvolusional dan lapisan *pooling* dianggap sebagai sebuah lapisan tunggal yaitu lapisan *convolutional-pooling*, maka *deep ConvNets* menggunakan lebih dari sebuah lapisan *convolutional-pooling*. Contoh arsitektur *deep ConvNets* dapat dilihat pada Gambar 3. Arsitektur tersebut menggunakan dua stage lapisan *convolutional-pooling* dan sebuah lapisan *fully-connected*.



Gambar 3. Contoh Arsitektur *Deep Convolutional Network* [8]

2. Metode Penelitian

Prosedur penelitian dibagi menjadi 3 tahap :

1. Tahap pembuatan data set
2. Tahap pembuatan model/pelatihan
3. Tahap pengujian

Tahap Pembuatan Data Set

ᩉ	ᩈ	ᩇ	ᩆ	ᩅ
ᩄ	ᩃ	ᩂ	ᩁ	ᩀ
ᩉ	ᩈ	ᩇ	ᩆ	ᩅ
ᩄ	ᩃ	ᩂ	ᩁ	ᩀ
ᩉ	ᩈ	ᩇ	ᩆ	ᩅ
ᩄ	ᩃ	ᩂ	ᩁ	ᩀ
ᩉ	ᩈ	ᩇ		

Gambar 4. Aksara Pallawa Hasil Pemindaian [6]

Citra tulisan tangan aksara Pallawa berjumlah 33 kelas. Sistem ini diharapkan mampu mengklasifikasikan 33 kelas tersebut. Setiap aksara Pallawa ditulis sebanyak 20 kali. Kemudian tulisan tersebut dipindai dengan format png. Ukuran citra diubah menjadi 28×28 pixel. Sehingga diperoleh sebanyak $33 \times 20 = 660$ data citra. Sebanyak 15 set data citra digunakan untuk proses pelatihan atau pembuatan model *deep Convnets*, sedangkan 5 set data citra digunakan untuk pengujian. Citra diubah dalam format *gray scale*. Contoh citra hasil pemindaian dapat dilihat pada Gambar 4.

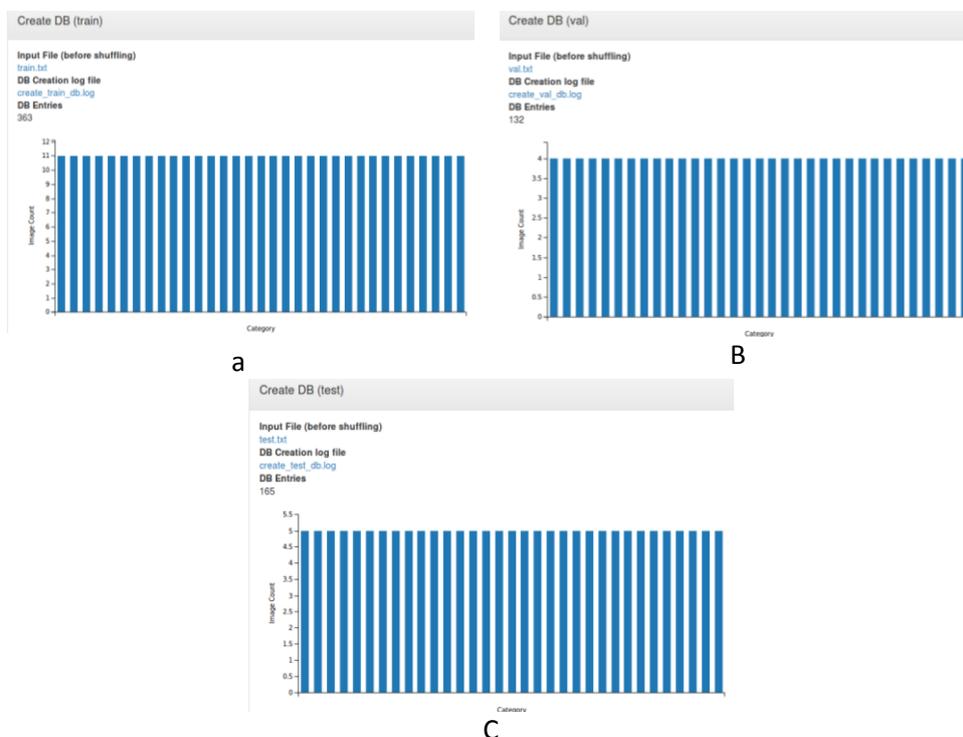
Tahap Pembuatan Model/Pelatihan

Pembuatan model *deep Convnets* atau tahap pelatihan merupakan tahap untuk menetapkan parameter-parameter yang akan digunakan untuk pengenalan. Pembuatan model menggunakan perangkat lunak DIGITS dari NVIDIA dengan sistem operasi Linux Ubuntu 16.04 LTS. GPU dengan CUDA core sejumlah 384 yang mendukung pemrosesan secara parallel digunakan untuk mempercepat proses pelatihan.

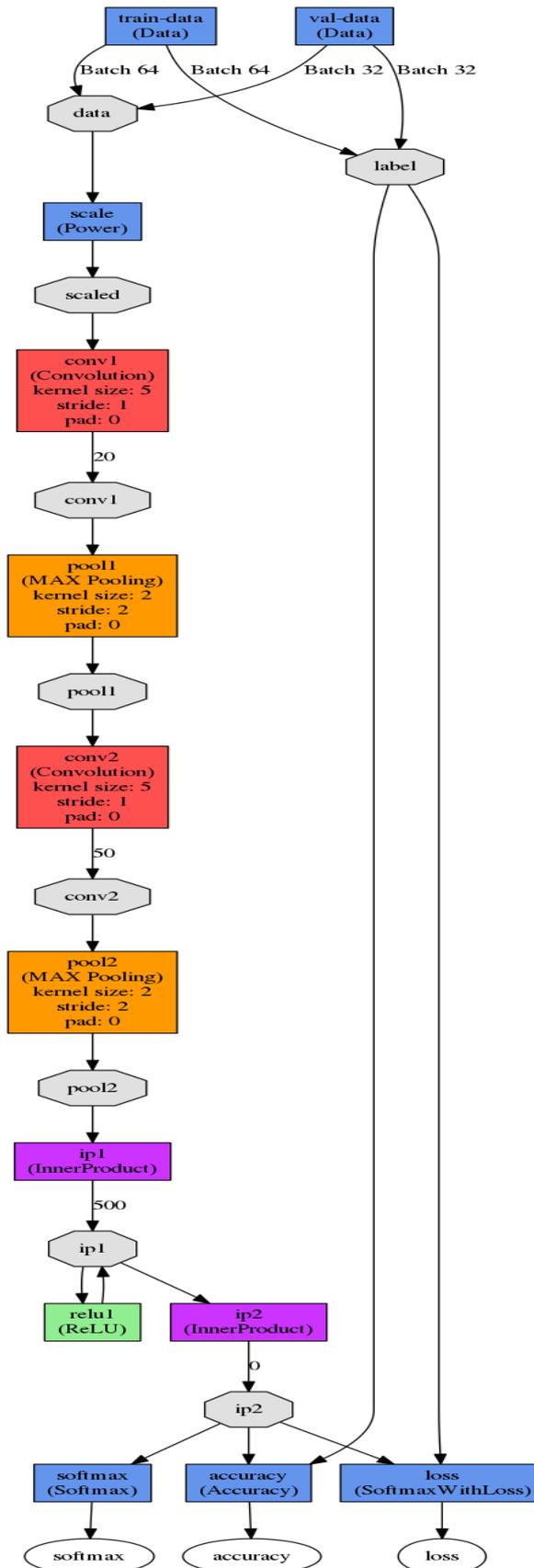
Model yang digunakan sebanyak 8 variasi. Sebanyak 4 model menggunakan metode *stochastic Gradient Descent* (*learning rate* awal=0.01), dengan variasi *epoch* sejumlah 10, 20, 30 dan 40 *epoch*. Empat model berikutnya menggunakan metode *adaptive gradient* (*learning rate* awal=0.01), dengan variasi *epoch* sejumlah 10, 20, 30 dan 40 *epoch*. Arsitektur dari *deep ConvNets* yang digunakan mengadopsi dari arsitektur LeNet dari Yann LeCun [9]. Diagram alir dari LeNet dapat dilihat pada Gambar 6. Lapisan yang digunakan adalah lapisan *convolution_1*, *max-pooling_1*, *convolution_2*, *max-pooling_2*, *fully-connected_1*, *fully-connected_2*.

Tahap Pengujian

Setelah memperoleh model pembelajaran, dengan jumlah total parameter pembelajaran sebanyak 442.602 parameter, tahap selanjutnya adalah tahap pengujian. Tahap ini merupakan proses mengklasifikasi citra uji menggunakan 8 variasi model pembelajaran.



Gambar 5. Hasil Pembuatan Data Set:
(a). Data Pelatihan, (b). Data Validasi, (c). Data Pengujian



Gambar 6. Diagram Alir Arsitektur LeNet (dari software DIGIT)

3. Hasil dan Pembahasan

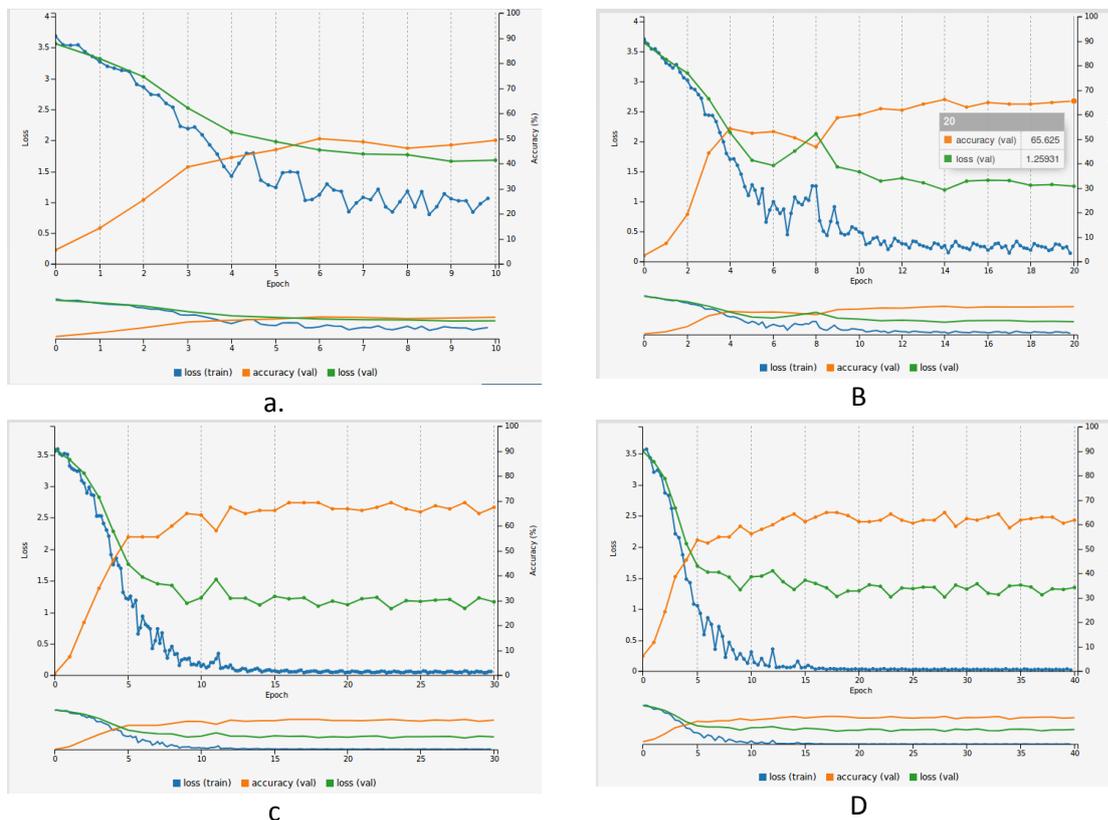
Hasil penelitian akan dibagi menjadi 3 bagian, yaitu pembuatan data set, tahap pembuatan model, dan tahap pengujian.

3.1. Hasil Pembuatan Data Set

Data set berjumlah 33 x 20 citra aksara Pallawa. Ukuran citra diubah menjadi 28 x 28 piksel dengan format *grayscale* dan disimpan dalam format *.png*. Hasil pembuatan data set tampak pada Gambar 5. Sebanyak 15 set citra digunakan untuk proses pelatihan dan validasi, 5 set digunakan untuk pengujian.

3.2. Hasil Pembuatan Model

Penelitian menghasilkan 8 jenis model, semuanya menggunakan arsitektur LeNet. Model yang pertama menggunakan metode *Stochastic Gradient Descent* (SGD) dengan *learning rate* awal=0.01, dengan variasi *epoch* sejumlah 10, 20, 30 dan 40 *epoch*. Grafik proses pembuatan model atau proses pelatihan Nampak pada Gambar 7. Empat model berikutnya menggunakan metode *Adaptive Gradient* (AG) dengan *learning rate* awal=0.01, dengan variasi *epoch* sejumlah 10, 20, 30 dan 40 *epoch*. Grafik proses pelatihan dapat dilihat pada Gambar 8.



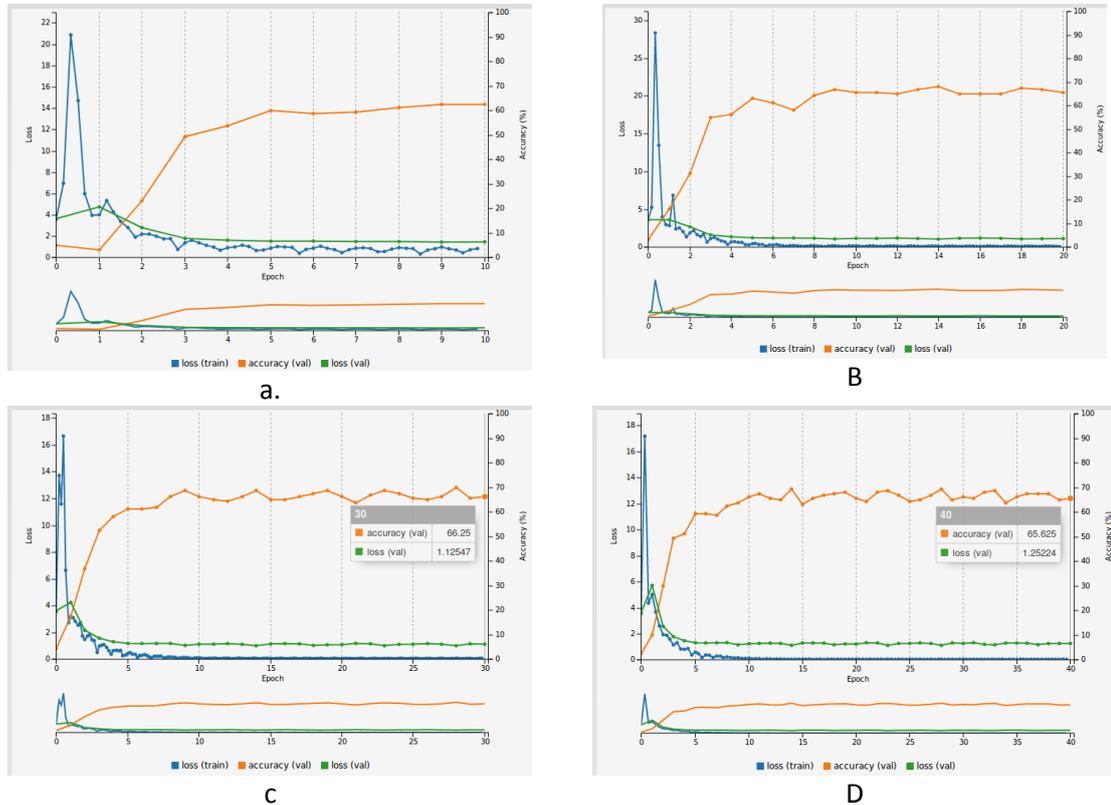
Gambar 7. Grafik Proses Pelatihan menggunakan *Stochastic Gradient Descent*:
(a). 10 epoch, (b). 20 epoch, (c). 30 epoch (d). 40 epoch

Tabel 1. Akurasi Model dan Waktu Pelatihan pada SGD

No.	Jumlah Epoch	Akurasi (%)	Waktu (detik)
1.	10	49,375	5
2.	20	65.625	6
3.	30	67,5	7
4.	40	61,875	8

Tabel keakuratan model dan waktu yang diperlukan saat pelatihan dengan SGD dapat dilihat pada Tabel 1. Sedangkan metode AG dapat dilihat pada tabel 2. Keakuratan ini diperoleh berdasarkan data set validasi.

Dari Tabel 1 tersebut dapat dilihat bahwa pelatihan dengan menggunakan jumlah epoch 30 menunjukkan hasil akurasi terbaik yaitu 67,5% dengan waktu yang diperlukan untuk pelatihan selama 7 detik. Dengan penambahan jumlah *epoch* ternyata justru menurunkan akurasi.



Gambar 8. Grafik Proses Pelatihan menggunakan *Adaptive Gradient*
(a). 10 *epoch*, (b). 20 *epoch*, (c). 30 *epoch*, (d). 40 *epoch*

Tabel 2. Akurasi Model dan Waktu Pelatihan pada *Adaptive Gradient*

No.	Jumlah <i>Epoch</i>	Akurasi (%)	Waktu (detik)
1.	10	62,5	5
2.	20	65.625	6
3.	30	66,25	7
4.	40	65,625	9

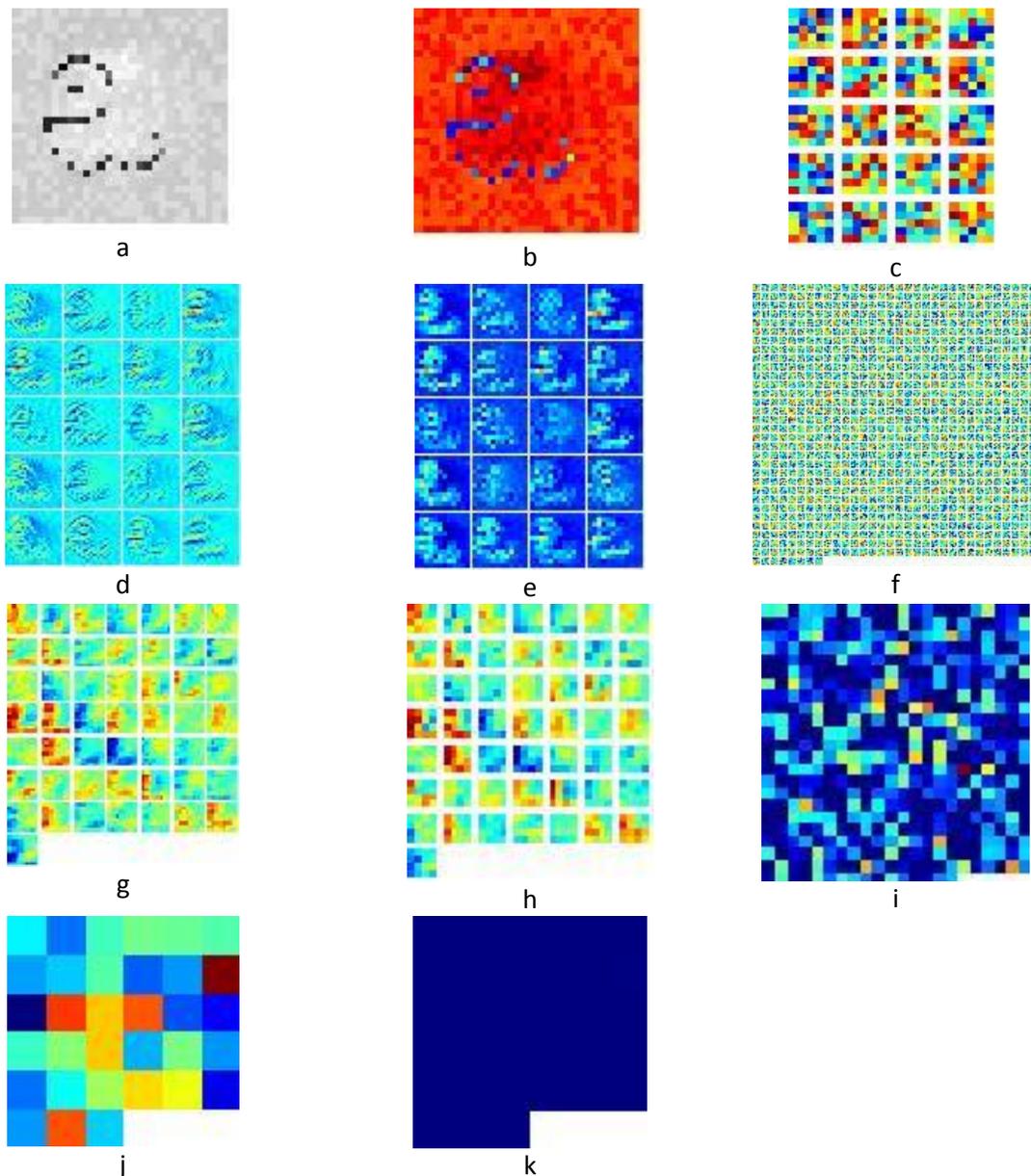
Dari Tabel 2 tersebut dapat dilihat bahwa pelatihan dengan menggunakan jumlah *epoch* 30 menunjukkan hasil akurasi terbaik yaitu 66,25% dengan waktu yang diperlukan untuk pelatihan selama 7 detik. Dengan penambahan jumlah *epoch* ternyata justru menurunkan akurasi. Sedangkan kalau dibandingkan antara metode SGD dan AG, akurasi terbaik diperoleh dengan metode SGD.

3.3. Hasil Pengujian

Contoh hasil pengujian menggunakan data tes tunggal dapat dilihat pada Gambar 8. Gambar 9 menyajikan proses dan hasil pengujian terhadap sebuah data aksara kelas 2 (aksara kha) pada model SGD 30 *epoch* pada tiap lapisan. Hasil prediksi yang diperoleh :

- Kelas 2 (aksara kha) : 95,3%
- Kelas 21 (aksara pa) : 2,12%
- Kelas 8 (aksara ja) : 1,21%
- Kelas 23 (aksara bha) : 1,07%
- Kelas 22 (aksara pha) : 0,1%

Hasil tersebut menunjukkan bahwa model dapat mengklasifikasi/mengenali aksara kha dengan baik dan sangat optimis (95,3%)



Gambar 9. Proses Klasifikasi Aksara kelas 2 (kha) pada model SGD 30 epoch:

- (a). Data asli, (b). Diskalakan, (c). conv1(*weight*), (d). conv1(*aktivasi*), (e). pool1(*max-pooling*),
 (f). conv2(*weight*), (g). conv2(*aktivasi*), (h). pool2(*max-pooling*),
 (i). ip1(*inner-product*), (j). ip2 k.*softmax*

4. Kesimpulan

Berdasarkan hasil pengamatan, *Deep Convolution Network* dapat mengenali atau mengklasifikasi aksara Pallawa dengan akurasi terbaik sebesar 67,5 % menggunakan *Stochastic Gradient Descent* dengan 30 *epoch*. Untuk penelitian selanjutnya, akan dikembangkan model yang juga dapat mengenali aksara yang dirotasi dan ditranslasi yaitu dengan cara memperbanyak data set pelatihan dan mencoba berbagai model jaringan.

Daftar Pustaka

- [1] Aksara Pallawa. www.Wikipedia.org/wiki/Aksara_Pallawa.
- [2] Yann L, Yoshua B, Geoffrey H. Deep Learning. *Nature*. 2015; 521:436-444
- [3] Alex K, Ilya S, Geoffrey H. *ImageNet Classification with Deep Convolutional Neural Network*. Neural Information Processing Systems Conference. 2012
- [4] Pierre S, Soumith C, Yann L. *Convolutional Neural Networks Applied to House Numbers Digit Classification*. International Conference on Pattern Recognition (ICPR 2012). 2012.
- [5] Michael A.N., *Neural Networks and Deep Learning*. Determination Press. 2015.
- [6] Wiwien W. Pengenalan Aksara Pallawa Dengan Model Hidden Markov. RETII XI. Yogyakarta. 2016;XI:6.
- [7] Adeshpande. *A Beginner's Guide To Understanding Convolutional Neural-Networks*. 2017
- [8] Adrian R, Lenet – *Convolutional Neural Network in Python*. Deep Learning Tutorials. 2016.
- [9] Yan L, Bottou, Bengio, Haffner. *Gradient-based Learning Applied to Document Recognition*. Proc. IEEE. 1998; 86(11): 2278–2324.