

KLASIFIKASI BURUNG CENDRAWASIH MENGUNAKAN RECOGNITION CNN KERAS

Warnia Nengsih, Ardiyanto, Ayu Putri Lestari

^a Politeknik Caltex Riau, Jl mbansari No 1 Rumbai, Pekanbaru 28265, Indonesia

^b Politeknik Caltex Riau, Jl Umbansari No 1 Rumbai, Pekanbaru 28265, Indonesia

^c Politeknik Caltex Riau, Jl Umbansari No 1 Rumbai, Pekanbaru 28265, Indonesia

^{1*} warnia@pcr.ac.id ; ² ardiyanto@gmail.com; ³ ayu_putri@gmail.com

INFORMASI ARTIKEL	ABSTRAK
Diterima : Direvisi : Diterbitkan : Kata Kunci: SVM Tensor CNN Keras Klasifikasi	Klasifikasi merupakan bagian dari <i>predictive modelling</i> dan <i>supervised learning</i> . Metode ini digunakan untuk menentukan kelas data berdasarkan nilai sebelumnya. Dalam penyelesaian kasus tertentu terdapat berbagai metode klasifikasi dengan tingkat akurasi yang beragam. <i>Convolutional Neural Network (CNN)</i> merupakan bagian <i>Multilayer Perceptron (MLP)</i> untuk melakukan pengolahan data dua dimensi. CNN juga merupakan bagian dari <i>Deep Neural Network</i> dan pada objek citra. Dari beberapa sumber menyatakan untuk proses klasifikasi menggunakan citra kurang tepat diimplementasikan pada MLP ini. Tentunya akan berakibat ke akurasi dari metode tersebut dalam menangani kasus tertentu. Pada penelitian ini proses klasifikasi objek menggunakan recognition keras untuk mengetahui nilai akurasi metode menggunakan objek burung cendrawasih. Dari hasil penelitian ini dilakukan training model dengan menggunakan 10 epochs dengan nilai akurasi sebesar 0.0850 sementara loss value sebesar 2.5658.
Keywords: SVM Tensor CNN Keras Classification	Classification is part of predictive modeling and supervised learning. This method is used to determine the data class based on the previous value. In solving certain cases, there are various classification methods with varying degrees of accuracy. Convolutional Neural Network (CNN) is part of the Multilayer Perceptron (MLP) for processing two-dimensional data. CNN is also part of the Deep Neural Network and is applied to image objects. From several sources, it is stated that the classification process using images is not properly implemented in this MLP. Of course, this will result in the accuracy of the method in handling certain cases. In this study, the object classification process uses hard recognition to determine the accuracy value of the method using the bird of paradise object. From the results of this study, a training model was carried out using 10 epochs with an accuracy value of 0.0850 while a loss value of 2.5658.

I. Pendahuluan

Convolutional Neural Network (CNN) merupakan bagian dari Deep Neural Network dan bisa diimplementasikan untuk objek citra. Di samping itu CNN bagian dari pengembangan Multilayer Perceptron (MLP). Dari beberapa sumber menyatakan untuk proses klasifikasi menggunakan citra kurang tepat diimplementasikan pada MLP ini. Tentunya akan berakibat ke akurasi dari metode tersebut dalam menangani kasus tertentu.[1]

Klasifikasi objek citra akan memetakan data ke dalam kelas tertentu. Pada penelitian ini klasifikasi objek menggunakan recognition keras dengan output penelitian berupa nilai akurasi dan loss value. Sehingga dari hasil pengolahan diperoleh hasil..

Data yang digunakan pada penelitian ini menggunakan dataset berupa citra burung cendrawasih. Terdapat 13 jenis burung cendrawasih yang akan digunakan yaitu Astrapia, Cendrawasih Biru, Cendrawasih Merah, Cendrawasih Panji, Cendrawasih Botak, Cendrawasih Goldi, Cendrawasih Belah Rotan, Cendrawasih Mati-Kawat, Manucodia, Cendrawasih Kera, Cendrawasih Raja, Toowa Cemerlang, Bidadari Halmahera. Dataset bersumber dari google dengan total dataset dengan jumlah 578 citra.

II. Landasan Teori

1.1 Algoritma Convolutional Neural Network (CNN)

Convolutional Neural Network (CNN) adalah pengembangan dari Multilayer Perceptron (MLP) yang didesain untuk mengolah data dua dimensi. CNN termasuk dalam jenis Deep Neural Network karena kedalaman jaringan yang tinggi dan banyak diaplikasikan pada data citra. Pada kasus klasifikasi citra, LP kurang sesuai untuk digunakan karena tidak menyimpan informasi spasial dari data citra dan menganggap setiap piksel adalah fitur yang independen sehingga menghasilkan hasil yang kurang baik.

1.2 Classification Model

Klasifikasi data mining adalah penempatan objek-objek ke salah satu dari beberapa kategori yang telah ditetapkan sebelumnya[2]. Klasifikasi banyak digunakan untuk memprediksi kelas pada suatu label tertentu, yaitu dengan mengklasifikasi data (membangun model) berdasarkan training set dan nilai-nilai (label kelas) dalam mengklasifikasikan atribut tertentu dan menggunakannya dalam mengklasifikasikan data yang baru. Pohon keputusan biasanya digunakan untuk mendapatkan informasi untuk tujuan pengambilan sebuah keputusan. Pohon keputusan dimulai dengan sebuah root node (titik awal) yang digunakan oleh user untuk mengambil tindakan[3]. Berdasarkan node root ini, user memecahkan leaf node sesuai dengan algoritma decision tree.

1.3.1 OpenCV

OpenCV (Open Source Computer Vision) adalah library dari fungsi pemrograman untuk realtime visi komputer. OpenCV menggunakan lisensi BSD dan bersifat gratis baik untuk penggunaan akademis maupun komersial. OpenCV dapat digunakan dalam bahasa pemrograman C, C++, Python, Java, dan sebagainya. OpenCV dapat digunakan pada sistem operasi Windows, Linux, Android, iOS dan Mac OS. OpenCV memiliki lebih dari 2500 algoritma yang telah dioptimalkan. [4,5]

1.3.2 Tensorflow

TensorFlow adalah sebuah perpustakaan software sumber terbuka yang digunakan untuk mesin belajar dalam berbagai macam tugas pemahaman persepsi dan bahasa. TensorFlow adalah API generasi kedua, menggantikan DistBelief (API generasi pertama) [6,7,8] yang digunakan oleh 50 tim yang berbeda untuk penelitian dan pengembangan puluhan produk komersial milik Google, seperti Speech Recognition, Gmail, Google Image, dan Google Search.

1.3.3 Keras

Keras adalah library Machine Learning open source yang ditulis dengan python untuk neural network, dikembangkan untuk membuat penerapan model pembelajaran yang mendalam secepat dan semudah mungkin untuk penelitian dan pengembangan yang dirilis di bawah lisensi MIT. Library Keras menyediakan cara yang bersih dan nyaman untuk membuat berbagai model pembelajaran mendalam di atas Theano atau TensorFlow yang menyediakan dasar untuk penelitian dan pengembangan Deep Learning[7]. Keras berfokus pada prinsip-prinsip utamanya yang mencakup kemudahan penggunaan, modularitas, dan ekstensibilitas yang mudah dengan Python[9,10].

III. Metodologi Penelitian

Data yang digunakan pada penelitian ini menggunakan dataset berupa citra burung cendrawasih. Terdapat 13 jenis burung cendrawasih yang akan digunakan pada Penelitian ini. Untuk keperluan data visualisasi dan manipulasi terdapat beberapa library yang diperlukan, diantaranya yaitu Numpy, Pandas, Matplotlib dan Seaborn[11,12,13]

Untuk keperluan model selection terdapat beberapa library yang akan digunakan, diantaranya yaitu `train_test_split` dari library `sklearn` yang digunakan untuk pembagian training dan testing, `Kfold` dari library `sklearn` untuk mengevaluasi kinerja model atau algoritma, `accuracy_score`, `precision_score`, `recall_score`, `confusion_matrix`, `roc_curve`, `roc_auc_score` dari library `sklearn` untuk mendapatkan akurasi `GridSearchCV` dari library. `Sklearn` untuk mencari parameter dalam “grid” yang diberikan. `LabelEncoder` dari library `sklearn` untuk mengubah label data `ImageDataGenerator`. `Keras.preprocessing.image` untuk pemrosesan data yang akan digunakan.[14,15].

Pada library `CNN` terdapat beberapa sublibrary yang akan digunakan, diantaranya yaitu : `Dropout`, `Flatten`, `Activation` dari `Keras` `Conv2D`, `MaxPooling2D`, `BatchNormalization` dari `Keras`, `Tensorflow`

IV. Hasil dan Pembahasan

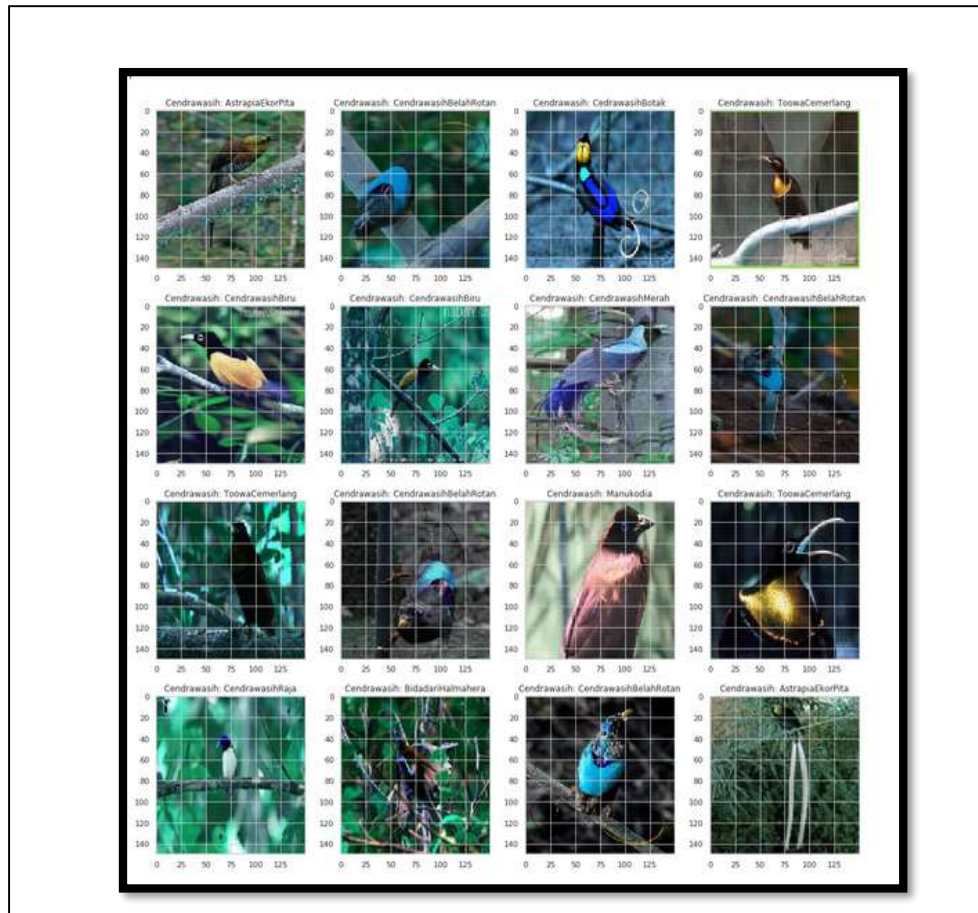
Proses import dataset yang akan digunakan pada kernel. Dataset yang digunakan adalah 13 jenis burung cendrawasih

Cendrawasih Biru (Blue Bird of Paradise) 	Cendrawasih Belah Rotan / Magnificent Bird of Paradise 	Toowa 
Cendrawasih Merah (Red Bird of Paradise) 	Twelve-wired Bird of Paradise 	Bidadari Halmahera / Semioptra Wallaci 
Cendrawasih Panji (King of Saxony Bird of Paradise) 	Manucodia 	Astrapia Ekor Pita / Ribbon Tailed Astrapia 
Cendrawasih Botak (Wilson's Bird of Paradise) 	Cendrawasih Kerah / Superb Bird of Paradise 	Cendrawasih Goldi (Goldi's Bird of Paradise) 
	Cendrawasih Raja / King Bird of Paradise 	

Gambar 1. Kategori dataset

Berdasarkan kategori dataset pada gambar 1 maka ada beberapa tahapan yang kita lakukan, diantaranya yaitu :Membuat array untuk menampung gambar beserta gambarnya. Membuat Variabel untuk memanggil masing-masing direktori folder untuk setiap jenis datanya.Membuat fungsi `assign_label()` yang digunakan untuk menyesuaikan label dengan jenis datasetnya. Tahapan selanjutnya yaitu membuat fungsi `make_train_data()` yang digunakan untuk mengolah data training. Proses ini akan terus berulang hingga semua dataset selesai di proses. Selanjutnya adalah menampilkan gambar dari dataset secara random.

Berikut adalah hasil dari visualisasi gambar secara random.

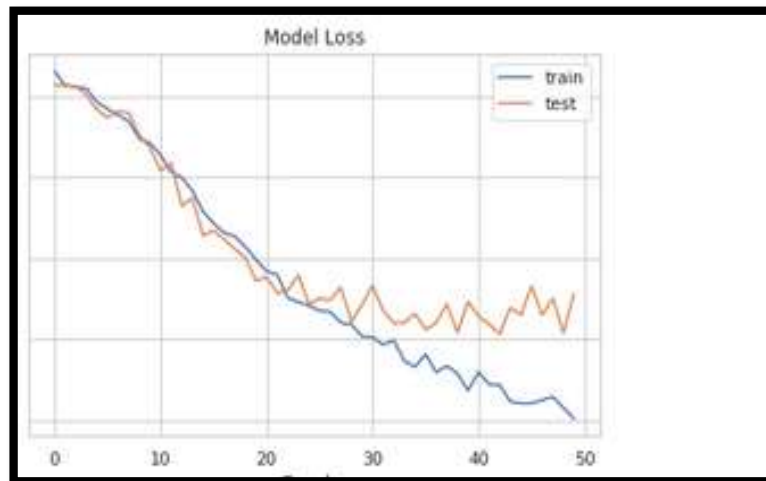


Gambar 2. Hasil Visualisasi Secara Random

Pada bagian ini bertujuan untuk pemberian label Y pada setiap jenis dataset yang ada dengan nomor, seperti astrapia = 0, Halmahera = 1, dan seterusnya sampai sebanyak jenis dataset. Berikutnya adalah membagi dataset menjadi data training dan data testing. Dimana pada bagian ini kita menentukan test_size sebesar 0.25 dan random statenya sebesar 42. Untuk mengimplementasikan model Convolutional Neural Network, beberapa hal menjadi perhatian seperti setting sequential, stride, flatten dan dense untuk menjalankan connection neural network.

Untuk penggunaan LR Annealer, setting konfigurasi untuk Batch_size adalah jumlah sample data yang disebar ke neural network. Epoch adalah ketika seluruh dataset sudah melalui proses training pada neural network sampai dikembalikan ke awal untuk sekali putaran, karena satu epoch terlalu besar untuk dimasukkan (feeding) kedalam komputer sehingga perlu membaginya kedalam satuan kecil (batches). Pada saat penyusunan modul keras dan summary default parameter untuk learning learning rate (Lr) $>=0$, parameter loss : digunakan untuk menentukan loss function, parameter metrics : digunakan untuk menentukan performa metric, selanjutnya menampilkan ringkasan dari modul Keras secara keseluruhan.

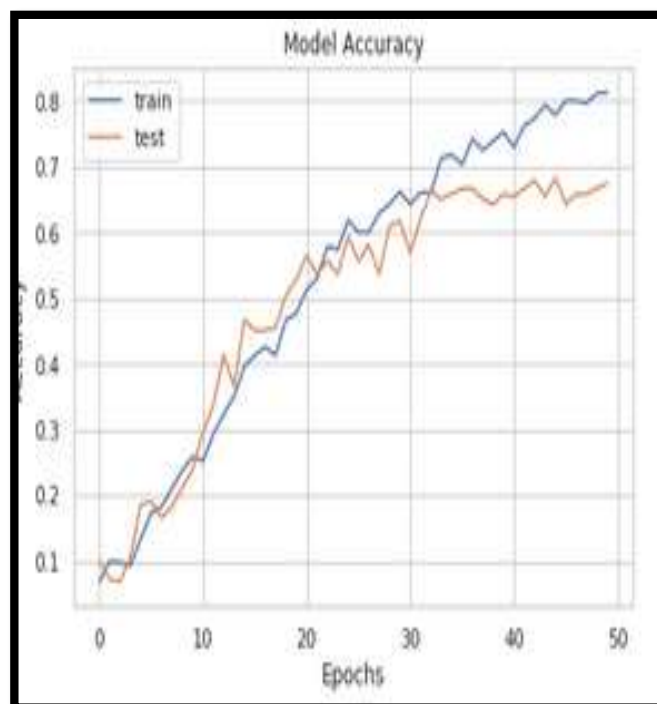
Tahapan selanjutnya melakukan prediksi untuk data testing yang ada dimana epochs yang digunakan sebanyak 50 epochs. Pada bagian Evaluasi Akurasi Model akan dilakukan pengecekan untuk menampilkan visualisasi ketidaktepatan antara data training dan data testing. Pada grafik berikut ini tersebut dapat dilihat bahwa grafik berbanding terbalik yang berarti dataset yang di analisis memiliki akurasi yang kecil.



Gambar 3 Grafik Evaluasi Model Loss

Selanjutnya dilakukan pengecekan untuk menampilkan visualisasi keakuratan antara data training dan testing.

Pada grafik tersebut dapat dilihat bahwa grafik berbanding lurus yang berarti dataset yang di analisis memiliki ketepatan yang tinggi.



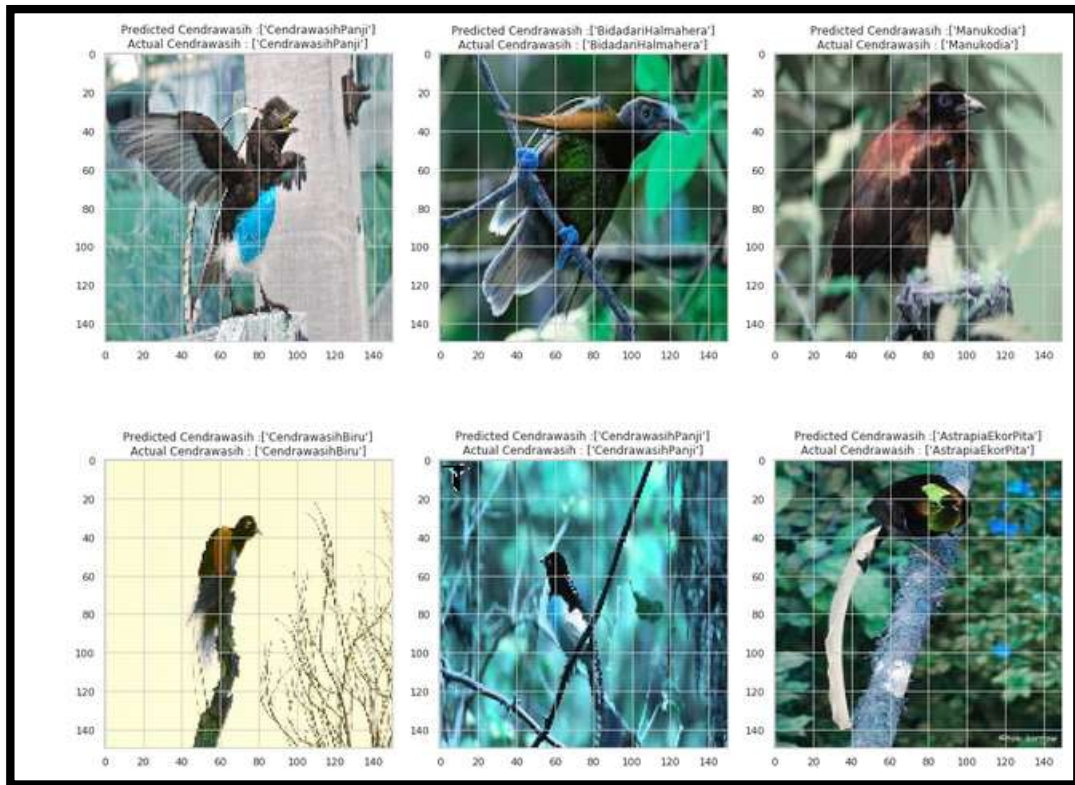
Gambar 4 Grafik Evaluasi Model Akurasi

Prediksi dilakukan untuk mengambil prediksi pada data testing. Selanjutnya akan dilakukan proses untuk melakukan antara klasifikasi yang benar dan yang salah.

Berikut menampilkan klasifikasi gambar yang benar atau sesuai prediksi dengan pengaturan:

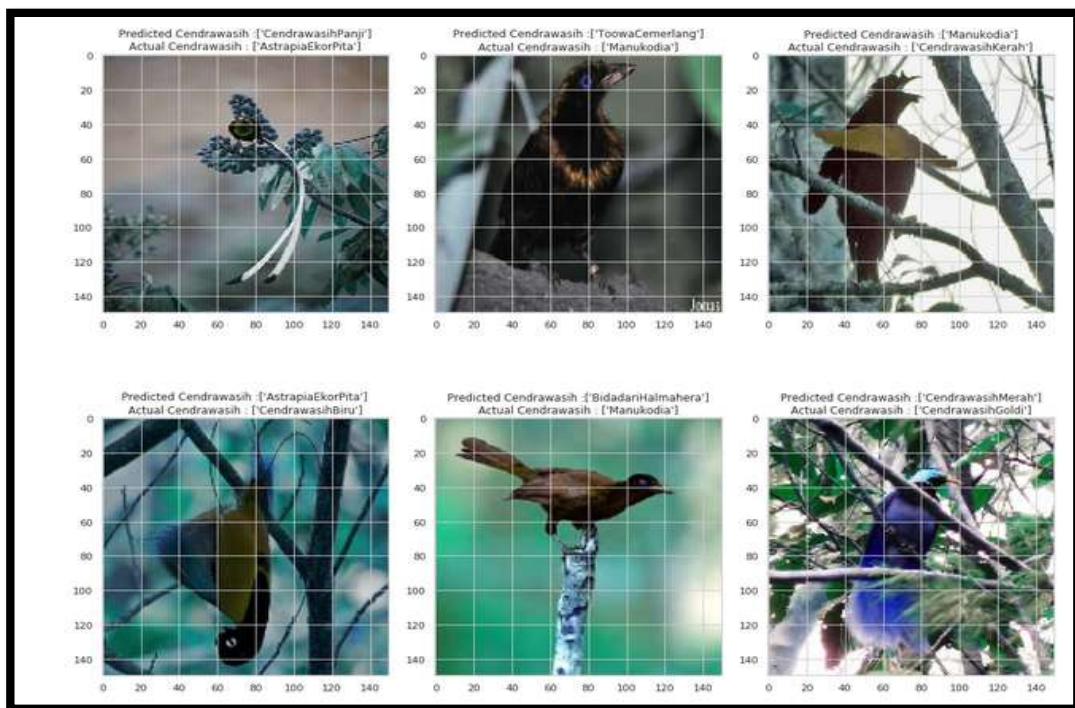
- Menampilkan 3 baris dengan 2 kolom dengan menggunakan looping
- Ukuran gambar / figure yaitu 15x15
- Menampilkan judul gambar yaitu prediksi dan actual

Hasil dari prediksi ini dapat dilihat pada gambar berikut ini



Gambar 5 Hasil Prediksi Data Testing Benar, figure 15x15

Berikut menampilkan klasifikasi gambar yang salah atau tidak sesuai prediksi.



Gambar 6 Hasil Prediksi Data Testing Salah, figure 15x15, menampilkan 2 baris dengan 3 kolom dengan menggunakan looping

Pada bagian ini dilakukan training model dengan menggunakan 10 epochs

```

In[15]: model.fit(train, labels, epochs=10)

Epoch 1/10
953/953 [=====] - 1s 543us/sample - loss: 4.4450 - acc: 0.0783
Epoch 2/10
953/953 [=====] - 0s 285us/sample - loss: 2.7348 - acc: 0.0934
Epoch 3/10
953/953 [=====] - 0s 290us/sample - loss: 2.5793 - acc: 0.0787
Epoch 4/10
953/953 [=====] - 0s 295us/sample - loss: 2.5779 - acc: 0.0672
Epoch 5/10
953/953 [=====] - 0s 301us/sample - loss: 2.5799 - acc: 0.0724
Epoch 6/10
953/953 [=====] - 0s 287us/sample - loss: 2.5871 - acc: 0.0783
Epoch 7/10
953/953 [=====] - 0s 288us/sample - loss: 2.5885 - acc: 0.0672
Epoch 8/10
953/953 [=====] - 0s 293us/sample - loss: 2.5702 - acc: 0.0860
Epoch 9/10
953/953 [=====] - 0s 287us/sample - loss: 2.5731 - acc: 0.0839
Epoch 10/10
953/953 [=====] - 0s 285us/sample - loss: 2.5658 - acc: 0.0850

Out[15]: <tensorflow.python.keras.callbacks.History at 0x7f1913c12978>

```

Gambar 7 Training Model

dapat dilihat pada gambar bahwa pada bagian iterasi ke 10 akurasi sebesar 0.0850 sementara loss sebesar 2.5658. Hal ini menunjukkan bahwa metode ini menunjukkan akurasi yang baik untuk objek citra.

V. Kesimpulan

Pada penelitian ini proses klasifikasi objek menggunakan recognition keras untuk mengetahui nilai akurasi metode menggunakan objek burung cendrawasih. Dari hasil penelitian ini dilakukan training model dengan menggunakan 10 epochs dengan nilai akurasi sebesar 0.0850 sementara loss value sebesar 2.5658..

Daftar Pustaka

Jenis Referensi	
Journal article	[1] Suartika, I. Y., Wijaya, A. Y., & Soelaiman, R. "Klasifikasi Citra Menggunakan Convolutional Neural Network (CNN)" Pada Caltech 101. <i>Jurnal Teknik Its</i> Vol. 5, No. 1.2016.
Book in Print	2] Meilina, P, <i>Penerapan Data Mining Dengan Metode Klasifikasi Menggunakan Decision Tree Dan Regresi</i> 2015.
Journal article	[3] He, K., X. Zhang, S. Ren and J. Sun. "Deep Residual Learning for Image Recognition, In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition". 770-778.2016.
Book in Print	[4] Lazaro, A., Bulial, J. L., & Amaliah, B. <i>Deteksi Jenis Kendaraan Di Jalan Menggunakan Opencv</i> . 6, 2017.
Journal article	[5] Simonyan, K. and A. Zisserman "Very deep convolutional networks for large-scale image recognition, arXiv: 1409.1556v.2015.
Journal article	[6] Krizhevsky, A., A. Sutskever and G. E. Hinton. "ImageNet Classification with Deep Convolutional Neural Networks". <i>Adv. Neural Info. Proc. Syst.</i> 25: 1097-1105, 2012.
Journal article	[7] Joseph, R., S. Divvala, R. Girshick and F.-h. Ali. "You only look once: Unified, Real-Time Object Detection, In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition", 779-788,2016.
	[8] Szegedy, C., W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke and A. Rabinovich. "Going Deeper with Convolutions, In

Jenis Referensi	
	Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition”, 1-9,2015.
Journal article	[9] Ciresan, D., Meier, U., Schmidhuber, J.: “Multi-column deep neural networks for im-age classification. In: Computer Vision and Pattern Recognition (CVPR)”, IEEEConference on. pp. 3642–3649. IEEE .2012
Journal article	[10] Ciresan, D.C., Meier, U., Masci, J., Maria Gambardella, L., Schmidhuber, J.: “Flexible,high performance convolutional neural networks for image classification”. In: IJCAIProceedings-International Joint Conference on Artificial Intelligence. vol. 22, p. 1237.2011.
Journal article	[11] Ciresan, D.C., Meier, U., Gambardella, L.M., Schmidhuber, J.: “Convolutional neuralnetwork committees for handwritten character classification. In: Document Analysisand Recognition (ICDAR), International Conference on. pp. 1135–1139. IEEE.2011.
Journal article	[12] Szegedy, C., Toshev, A., Erhan, D.: “Deep neural networks for object detection. In:Advances in Neural Information Processing Systems”. pp. 2553–2561. 2013.
Journal article	[13] Tivive, F.H.C., Bouzerdoun, A.: “A new class of convolutional neural networks(siconnets) and their application of face detection. In: Neural Networks”, Pro-ceedings of the International Joint Conference on. vol. 3, pp. 2157–2162. IEEE. 2003.
Journal article	[14] Zeiler, M.D., Fergus, R.: “Stochastic pooling for regularization of deep convolutionalneural networks”. arXiv preprint arXiv:1301.3557, 2013.
Journal article	[15] Zeiler, M.D., Fergus, R.: “Visualizing and understanding convolutional networks”. In:Computer Vision–ECCV 2014, pp. 818–833. Springer .2014.