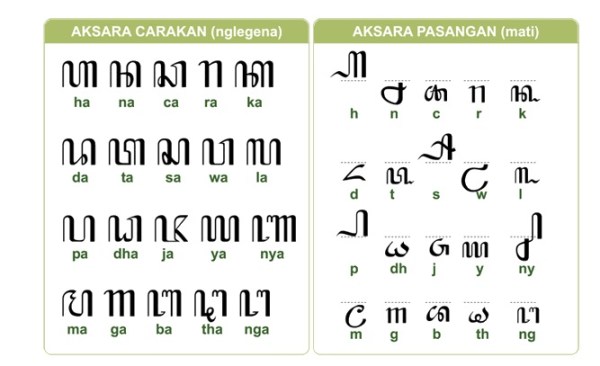
# **BAB II TINJAUAN PUSTAKA**

Bab ini menjelaskan tentang definisi dan teori-teori yang digunakan sebagai landasan penelitian yang berasal dari hasil publikasi dan penelitian dan/atau buku yang relevan.

## **AKSARA JAWA**

Aksara Jawa, yang juga dikenal sebagai Hanacaraka dan Dentawyanjana, adalah salah satu aksara tradisional di Indonesia yang berkembang di Pulau Jawa. Aksara ini terutama digunakan untuk menulis bahasa Jawa, tetapi juga digunakan untuk menulis beberapa bahasa daerah lainnya seperti bahasa Sunda, Madura, Sasak dan Melayu serta bahasa historis seperti Sanskerta dan Kawi. Aksara Jawa berasal dari aksara Brahmi India melalui aksara Kawi dan berkerabat dekat dengan aksara Bali. Aksara ini aktif digunakan dalam sastra dan tulisan sehari-hari oleh masyarakat Jawa dari pertengahan abad ke-15 hingga pertengahan abad ke-20 sebelum perlahan-lahan digantikan oleh huruf Latin. Aksara ini masih diajarkan di DI Yogyakarta, Jawa Tengah, Jawa Timur, Cirebon dan Indramayu sebagai bagian dari muatan lokal, tetapi penggunaannya terbatas dalam kehidupan sehari-hari. Aksara Jawa merupakan sistem tulisan *abugida* yang terdiri dari sekitar 20 aksara dasar (Poerwadarminta, 1939).



Gambar 1 Aksara Carakan  
sumber: https://www.rukita.co/stories/aksara-jawa-lengkap/

## **CITRA DIGITAL**

Citra digital terdiri dari sejumlah elemen terbatas, masing-masing memiliki lokasi dan nilai tertentu. Elemen-elemen ini disebut elemen citra, pel, dan piksel. Sebuah citra digital dapat didefinisikan sebagai fungsi dua dimensi, f(x,y), di mana x dan y adalah koordinat spasial (bidang), dan amplitudo f pada setiap pasangan koordinat (x,y) disebut intensitas atau tingkat abu-abu citra di titik tersebut. Bidang pengolahan citra digital mengacu pada pengolahan citra digital dengan menggunakan komputer digital (Gonzalez & Woods, 2018)

## **PENGOLAHAN CITRA DIGITAL**

Pemrosesan citra adalah kumpulan teknik komputasi untuk menganalisis, meningkatkan, mengompres, dan merekonstruksi citra. Komponen utamanya meliputi impor, di mana sebuah citra ditangkap melalui pemindaian atau fotografi digital; analisis dan manipulasi citra yang dilakukan menggunakan berbagai aplikasi perangkat lunak khusus; serta output (misalnya, ke printer atau monitor). Pemrosesan citra memiliki berbagai aplikasi yang luas di berbagai bidang, termasuk astronomi, kedokteran, robotik industri, dan pemantauan jarak jauh oleh satelit (Gonzalez & Woods, 2018).

## **RESCALE/RESIZE**

Proses *rescale/resize* pada citra merupakan tahap pra-pemrosesan yang kritis dalam visi komputer. Secara prinsip, model *deep learning* dapat dilatih lebih cepat pada citra berukuran kecil. Citra input yang lebih besar memerlukan jaringan saraf untuk belajar dari empat kali lipat jumlah piksel, yang pada akhirnya meningkatkan waktu pelatihan untuk arsitektur tersebut (Saponara & Elhanashi, 2022).

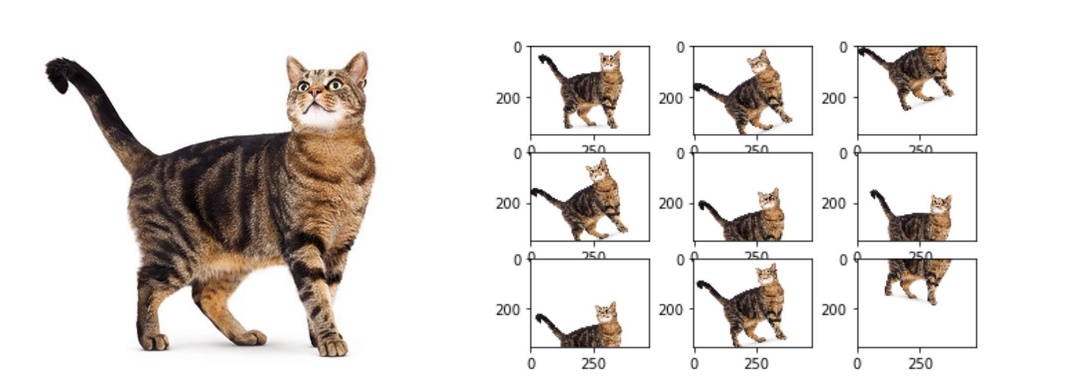
A picture containing diagram, line

Description automatically generated

Gambar 2 Rescale Bilinear Interpolation  
sumber: https://theailearner.com/2018/12/29/image-processing-bilinear-interpolation/

## **AUGMENTASI DATA**

Tujuan dari augmentasi data adalah menambahkan titik data baru ke ruang input dengan memodifikasi citra pelatihan sambil mempertahankan informasi semantik dan label target. Oleh karena itu, augmentasi data digunakan untuk mengurangi *overfitting*. Beberapa penelitian mengkonfirmasi pentingnya augmentasi data baik dalam pelatihan maupun pengujian, dan menunjukkan bahwa hal tersebut dapat menghasilkan peningkatan kinerja yang lebih besar daripada mengumpulkan citra sebagai dataset baru (Perez et al., 2018).



Gambar 3 Augmentasi Data  
sumber: https://analyticsindiamag.com/image-data-augmentation-impacts-performance-of-image-classification-with-codes/

## **NORMALISASI dan STANDARDISASI**

Normalisasi dan standarisasi citra adalah proses yang digunakan dalam pemrosesan citra untuk memastikan perbandingan yang optimal antara metode akuisisi data dan instansi tekstur. Tujuan utama dari normalisasi dan standarisasi citra adalah mengubah piksel citra sehingga mereka dapat dibandingkan secara konsisten dalam berbagai situasi (Gonzalez & Woods, 2018).

(2.1)

(2.2)

## **DATA SPLITTING**

*Data splitting* atau pemisahan data melibatkan pembagian dataset menjadi tiga bagian: set pelatihan, set validasi, dan set pengujian. Set pelatihan digunakan untuk membangun model, sementara set validasi dan pengujian berfungsi sebagai set penahan. Tidak ada proporsi yang pasti untuk pembagian dataset ini, namun aturan umum sebelumnya adalah 70% untuk pelatihan dan 15% untuk validasi dan pengujian. Namun, dalam era *big data*, proporsi seperti 95% untuk pelatihan dan 2,5%/2,5% untuk validasi/pengujian dapat dipertimbangkan. Dengan menggunakan ketiga subset ini, diharapkan model yang dibangun dapat memprediksi dengan baik contoh-contoh yang tidak dilihat oleh algoritma pembelajaran. Set validasi digunakan untuk memilih algoritma pembelajaran dan menentukan nilai *hyperparameter* terbaik, sementara set pengujian digunakan untuk mengevaluasi model sebelum digunakan di lapangan (Burkov, 2019).

## **DEEP LEARNING**

*Deep learning* adalah cabang dari *machine learning* yang menggunakan jaringan saraf tiruan dengan banyak lapisan tersembunyi untuk memodelkan data yang kompleks (Lecun et al., 2015). Teknik ini memungkinkan komputer untuk mempelajari representasi data yang abstrak dan hierarkis secara otomatis dari data mentah, tanpa perlu pengetahuan domain yang spesifik atau rekayasa fitur yang rumit. *Deep learning* telah mencapai hasil yang sangat baik dalam berbagai tugas pengenalan pola, termasuk pengenalan suara, pengenalan gambar, dan pemrosesan bahasa alami.

## **CONVOLUTIONAL NEURAL NETWORK**

*Convolutional Neural Network* (CNN) adalah jenis *network* khusus yang secara signifikan mengurangi jumlah parameter dalam *deep neural* *network* dengan banyak unit tanpa kehilangan terlalu banyak kualitas model. CNN telah menemukan aplikasi dalam pengolahan gambar dan teks di mana mereka mengalahkan banyak patokan yang sebelumnya ditetapkan (Burkov, 2019). Dalam implementasi *transfer learning*, CNN bertugas sebagai mengekstrak fitur dari citra data pada model tersebut.

## **ACTIVATION FUNCTIONS**

Untuk memungkinkan jaringan saraf mempelajari batas keputusan yang kompleks, kami menerapkan fungsi aktivasi non-linier pada beberapa lapisannya. Fungsi yang umum digunakan meliputi *tanh, ReLU, softmax*, dan varian dari fungsi-fungsi tersebut. Secara teknis, setiap neuron menerima sinyal masukan yang merupakan jumlah terbobot dari bobot sinaptik dan nilai aktivasi dari neuron yang terhubung (Zaccone & Karim, 2018).

(3)

(4)

## **DROPOUT**

*Dropout* adalah teknik yang digunakan untuk mengurangi *overfitting* dalam jaringan dengan banyak lapisan dan/atau neuron. Secara umum, lapisan *dropout* ditempatkan setelah lapisan yang memiliki sejumlah besar neuron yang dapat dilatih (Zaccone & Karim, 2018).

Teknik ini mengubah pendekatan pembelajaran bobot. Alih-alih mempelajari semua bobot jaringan bersama-sama, *dropout* melatih sebagian dari mereka dalam iterasi pelatihan *batch* (Ranjan, 2019).

**POOLING (TBA)**

## **TRANSFER LEARNING**

*Transfer learning* adalah peningkatan pembelajaran dalam tugas baru melalui transfer pengetahuan dari tugas terkait yang sudah dipelajari (Torrey & Shavlik, 2009). *Transfer learning* terdiri dari mengambil jaringan yang sudah dibangun dan membuat perubahan yang sesuai pada parameter dari berbagai lapisan sehingga dapat menyesuaikan dengan dataset lain (Zaccone & Karim, 2018).

**A picture containing text, screenshot, font, diagram

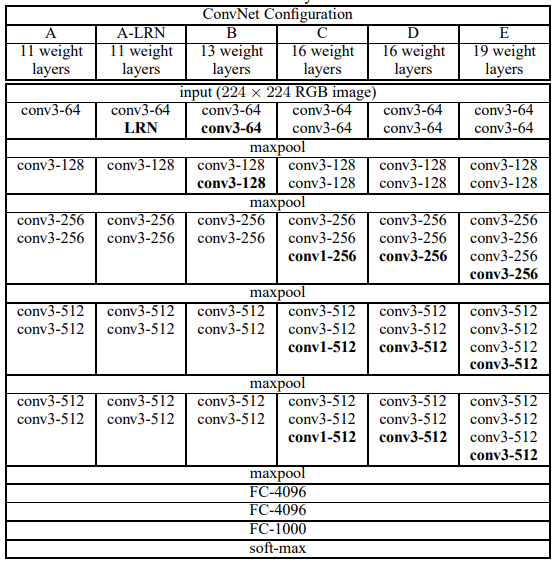
Description automatically generated**

Gambar 4 Most Common Deep Transfer Learning approaches.  
(Iman et al., 2023)

Pendekatan populer pertama adalah menyesuaikan model yang telah dilatih pada data target, hal ini merupakan metode *Deep Transfer Learning* (DTL) yang paling umum digunakan karena kemudahannya. Metode ini dapat meningkatkan pelatihan pada data target dengan mengurangi biaya pelatihan dan mengatasi kebutuhan dataset target yang luas, tetapi masih rentan terhadap pelupakan yang mematikan. Metode ini telah diterapkan pada dataset gambar dan tabular di berbagai bidang. Pendekatan populer kedua adalah membekukan lapisan CNN dalam model yang telah dilatih dan menyesuaikan hanya lapisan terhubung sepenuhnya lateral. Dalam metode ini, lapisan CNN mengekstrak fitur dari dataset yang diberikan, dan *fully connected layer* bertanggung jawab untuk klasifikasi dan akan disesuaikan dengan tugas baru untuk data target (Iman et al., 2022).

## **VGG Pre-trained Model**

Visual Geometric Group (VGG) adalah model arsitektur jaringan saraf tiruan yang sangat dalam dan dikembangkan oleh Visual Geometry Group di Universitas Oxford (Simonyan & Zisserman, 2014). Model ini terdiri dari beberapa lapisan konvolusi yang diikuti oleh lapisan *max-pooling* dan beberapa lapisan sepenuhnya terhubung di bagian akhir. Salah satu keunggulan utama dari model VGG adalah penggunaan filter konvolusi berukuran kecil (3x3) di semua lapisannya, yang memungkinkan jaringan untuk memiliki kedalaman yang lebih besar dengan jumlah parameter yang sama dibandingkan dengan jaringan yang menggunakan filter konvolusi berukuran lebih besar. Model VGG mencapai hasil yang sangat baik pada *benchmark* klasifikasi ILSVRC 2012 dan telah digunakan secara luas sebagai titik awal untuk berbagai tugas pengenalan gambar.

****

Gambar 5 Arsitektur VGG  
(Simonyan & Zisserman, 2014)

## **INCEPTION Pre-trained model**

Inception adalah model arsitektur jaringan saraf tiruan yang digunakan untuk visi komputer. Model ini dirancang untuk meningkatkan kualitas jaringan dengan cara yang efisien secara komputasi dengan menggunakan konvolusi yang difaktorkan dan regularisasi agresif (Szegedy et al., 2015). Inception-v3 adalah salah satu versi dari model Inception yang mencapai kinerja tinggi pada *benchmark* klasifikasi ILSVRC 2012 dengan biaya komputasi yang relatif rendah dibandingkan dengan arsitektur yang lebih sederhana dan monolitik (Szegedy et al., 2016). Model ini juga menggunakan jumlah parameter yang lebih sedikit dan regularisasi tambahan dengan *classifier auxilary* yang dinormalisasi *batch* dan *label-smoothing* untuk melatih jaringan berkualitas tinggi pada set pelatihan berukuran relatif sedang (Szegedy et al., 2016).

**A picture containing text, screenshot, number, font

Description automatically generated**

Gambar 6 Arsitektur Inception  
(Szegedy et al., 2016)

## **XCEPTION Pre-trained model**

Xception adalah arsitektur jaringan saraf konvolusional yang sepenuhnya didasarkan pada lapisan konvolusi terpisah secara mendalam *(depthwise separable convolution layers).* Arsitektur ini diusulkan sebagai perpanjangan dari arsitektur Inception, di mana modul Inception telah digantikan dengan konvolusi terpisah secara mendalam. Arsitektur Xception adalah tumpukan linier lapisan konvolusi terpisah secara mendalam dengan koneksi residual (Chollet, 2016).

**A screenshot of a computer program

Description automatically generated with low confidence**

Gambar 7 Arsitektur Xception  
(Chollet, 2016)

## **CONFUSION MATRIX**

Confusion Matrix adalah tabel yang merangkum seberapa sukses model klasifikasi dalam memprediksi contoh yang termasuk dalam berbagai kelas. Salah satu sumbu dari *confusion matrix* adalah label yang diprediksi oleh model, dan sumbu lainnya adalah label sebenarnya. Dalam masalah klasifikasi biner, terdapat dua kelas (Burkov, 2019).

## **ALBUMENTATIONS**

Albumentations adalah *library* Python untuk augmentasi citra yang cepat dan fleksibel. *Library* ini secara efisien mengimplementasikan berbagai operasi transformasi gambar yang kaya dan dioptimalkan untuk kinerja, sambil memberikan antarmuka augmentasi gambar yang ringkas namun kuat untuk berbagai tugas visi komputer, termasuk klasifikasi objek, segmentasi, dan deteksi (Buslaev et al., 2020).

## **TENSORFLOW**

TensorFlow adalah antarmuka untuk mengekspresikan algoritma *machine learning* dan implementasi untuk menjalankan algoritma tersebut. Sistem ini fleksibel dan dapat digunakan untuk mengekspresikan berbagai macam algoritma. TensorFlow telah digunakan untuk penelitian dan penerapan sistem pembelajaran mesin di berbagai bidang. API TensorFlow dan implementasi referensi dirilis sebagai paket sumber terbuka pada November 2015 dan tersedia di [www.tensorflow.org](http://www.tensorflow.org) (Abadi et al., 2016).

## **REVIEW LITERATUR**

|  |  |  |  |
| --- | --- | --- | --- |
| **Judul** | **Penulis, Tahun** | **Metode** | **Hasil** |
| *Transfer Learning Implementation on Sundanese Script Recognition Using Convolutional Neural Network* | MUHAMMAD KHALIFA U, et al. (2022) | *Transfer learning* dengan *pre-trained* model dari dataset *alphabet*, *devanagari*, *arabic*, dan aksara jawa | Model *transfer learning* terbaik didapat dari *pre-trained* dataset *arabic,* mencapai akurasi 91,86% dan *loss* 0.2814 |
| Javanese Script Text Image Recognition Using Convolutional Neural Networks | Goldy Najma Adli Kesaulya, et al. (2022) | *Transfer learning* dengan *pre-trained* model ResNeXt dengan melakukan *freezing* di 4 layer pertama dari 10 layer model *pre-trained* | Akurasi testing mencapai 98.19% |
| Deep Learning for Image Classification on Very Small Datasets Using Transfer Learning | Menying Shu (2019) | *Fine-tuned pre-trained* model VGG16, VGG19, InceptionV3, InceptionResNetV2 | Akurasi testing mencapai 96% dengan menggunakan *fine-tuned* model dari InceptionResNetV2 |
| A Close Look at Deep Learning with Small Data | L Brigato, et al. (2020) | Review metode yang dapat digunakan untuk menyelesaikan permasalahan *small data*, menggunakan berbagai variasi dari model CNN dan ResNet | *Regularization technique* seperti *data augmentation* dan dropout dapat meningkatkan performa dari *deep learning* |
| A Comprehensive Study on Deep Image Classification with Small Datasets | Gayani Chandrarathne, et al. (2019) | Review metode *transfer learning* dengan *fine-tuned pre-trained* model VGG-16 pada berbagai macam dataset. | *Fine-tuning* (*re-init few layers*)dan *(whole network)* menunjukkan performa yang signifikan dibandingkan dengan *scratch training.* Hasil akurasi yang diperoleh dari *fine-tuning* adalah 91.4%, 95.52% dan 79.6% secara terurut |
| Understanding the Mechanisms of Deep Transfer Learning for Medical Images | Hariharan Ravishankar, et al. (2017) | CNN *transfer learning* CaffeNet | *Transferred* dan *fine-tuned* model dapat menungguli performa *state-of-the-art* *feature engineered pipeline* (Haar) dan menghasilkan akurasi 85% |
| Transfer Learning using CNN for Handwritten Devanagari Character Recognition | Nagender Aneja, et al. (2019) | *Transfer learning* AlexNet, DenseNet, VGG, dan Inception | InceptionV3 menghasilkan akurasi sebesar 99% |
| A Deep Learning-Based Framework for Automatic Brain Tumors Classification Using Transfer Learning | Arshia Rehman, et al. (2019) | *Transfer learning* menggunakan *pre-trained* model AlexNet, GoogLeNet, dan VGG16 dari dataset ImageNet | *Fine-tuned* VGG16 menghasilkan akurasi tertinggi sebesar 98.69% |
| A Review of Deep Transfer Learning and Recent Advancements | Mohammadreza Iman, et al. (2023) | Review berbagai metode *deep transfer learning* yang sudah diteliti dalam waktu dekat | (i) *Finetuning*, (ii) *Freezing* CNN *Layers*, dan (iii) *Progressive Learning* adalah teknik yang telah terbukti kemampuan dan efektivitasnya untuk berbagai masalah *machine learning*. |
| Text recognition on images using pre-trained CNN | Afgani Fajar Rizky, et al. (2023) | *Transfer leraning* dengan *augmentation, freeze-layers,* dan *fine-tuning* telah dilakukan dengan menggunakan model AlexNet, VGG, ResNet, dan DenseNet | Akurasi terbaik dihasilkan dari model VGG dengan 0 *freeze layer* dan *augmentation* |
| Transfer learning using VGG-16 with Deep Convolutional Neural Network for Classifying Images | Srikanth Tammina (2019) | *Transfer learning* menggunakan *augmentation, fine-tuning,* dan *freeze layers* dengan *pre-trained* model VGG | Hasil terbaik diperoleh dari proses *fine-tuning* dan *augmentation* dengan akurasi 95.40% pada validation |

**DAFTAR PUSTAKA**

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., … Research, G. (2016). *TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems*. https://arxiv.org/abs/1603.04467v2

Buslaev, A., Iglovikov, V. I., Khvedchenya, E., Parinov, A., Druzhinin, M., & Kalinin, A. A. (2020). Albumentations: Fast and flexible image augmentations. *Information (Switzerland)*, *11*(2). https://doi.org/10.3390/INFO11020125

Chollet, F. (2016). Xception: Deep Learning with Depthwise Separable Convolutions. *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, *2017-January*, 1800–1807. https://doi.org/10.1109/CVPR.2017.195

Iman, M., Rasheed, K., & Arabnia, H. R. (2022). A Review of Deep Transfer Learning and Recent Advancements. *Technologies*, *11*(2), 40. https://doi.org/10.3390/technologies11020040

Lecun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature 2015 521:7553*, *521*(7553), 436–444. https://doi.org/10.1038/nature14539

Perez, F., Vasconcelos, C., Avila, S., & Valle, E. (2018). Data Augmentation for Skin Lesion Analysis. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *11041 LNCS*, 303–311. https://doi.org/10.1007/978-3-030-01201-4\_33

Saponara, S., & Elhanashi, A. (2022). Impact of Image Resizing on Deep Learning Detectors for Training Time and Model Performance. *Lecture Notes in Electrical Engineering*, *866 LNEE*, 10–17. https://doi.org/10.1007/978-3-030-95498-7\_2/COVER

Simonyan, K., & Zisserman, A. (2014). Very Deep Convolutional Networks for Large-Scale Image Recognition. *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*. https://arxiv.org/abs/1409.1556v6

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., & Rabinovich, A. (2015). Going deeper with convolutions. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, *07-12-June-2015*, 1–9. https://doi.org/10.1109/CVPR.2015.7298594

Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., & Wojna, Z. (2016). Rethinking the Inception Architecture for Computer Vision. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, *2016-December*, 2818–2826. https://doi.org/10.1109/CVPR.2016.308

# **DAFTAR PUSTAKA BUKU**

Burkov, A. (2019). *The Hundred-Page Machine Learning Book.* Andriy Burkov.

Gonzalez, R. C., & Woods, R. E. (2018). *Digital Image Processing (4th ed., Global ed.).* Harlow, England: Pearson Education Limited.

Ranjan, C. (2019). *Understanding Deep Learning: Application in Rare Event Prediction.* Chitta Ranjan.

Torrey, L., & Shavlik, J. (2009). *Chapter: Transfer Learning.* Hershey, Pennsylvania: IGI Global.

Zaccone, G., & Karim, M. R. (2018). *Deep Learning with TensorFlow: Explore neural networks and build intelligent systems with Python (2nd ed.).* Birmingham, United Kingdom: Packt Publishing.