BAB IV

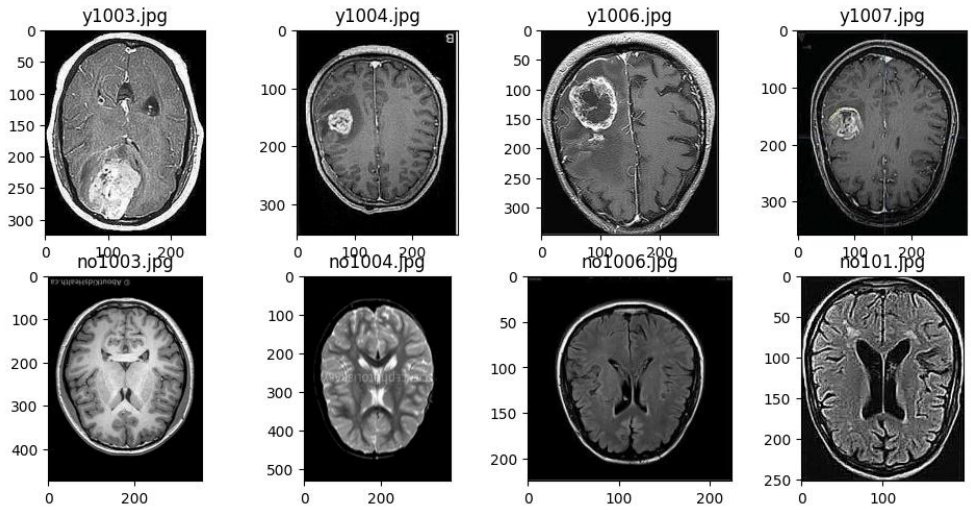
HASIL DAN PEMBAHASAN

4.1 Hasil

Berdasarkan metodologi yang telah dirancang pada kasus deteksi tumor otak pada gambar MRI dengan metode *Deep Neural Network* didapatkan hasil sebagai berikut.

a. ***Exploratory Data Analysis***

Data yang telah dikumpulkan melalui platform repositori *Kaggle* berjumlah 3000 gambar otak yang merupakan hasil pindaian *Magnetic Resonance Imaging* dengan orientasi *axial*. Gambar tersebut dilihat sebaran gambarnya dengan dipilih secara acak lalu ditampilkan sekaligus dengan tata letak 4 x 2 kolom yang dapat dilihat pada gambar 4.1.



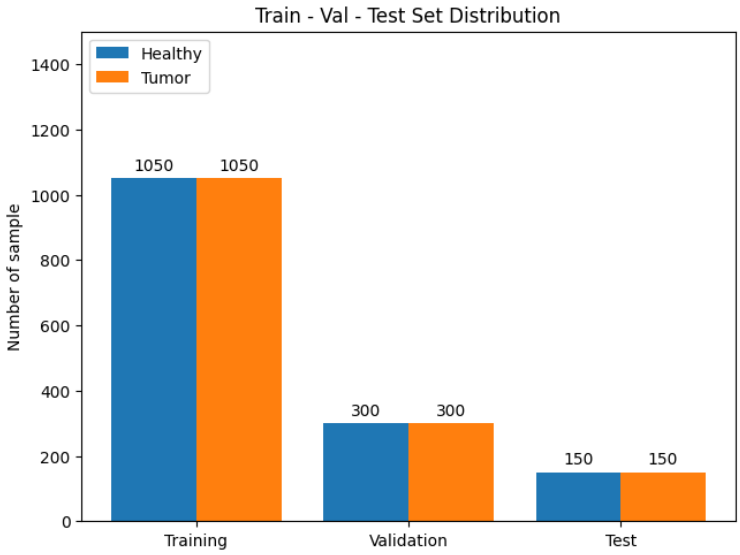
Gambar 4.1 Tampilan sebaran gambar otak pada dataset

Berdasarkan gambar 4.1 diatas dapat dilihat bahwa ukuran gambar pada dataset relatif beragam dengan kisaran ukuran 200 x 250 piksel sampai dengan ukuran 500 x 400 piksel.

Gambar terdiagnosis tumor terlihat memiliki sebuah pola menyerupai lingkaran didalam area otak yang mencirikan abnormalitas otak. Bentuk abnormalitas tersebut nantinya akan menjadi fitur utama dalam memprediksi keberadaan tumor dalam otak manusia yang diekstrak melalui lapisan konvolusi / *layers Convolutional*.

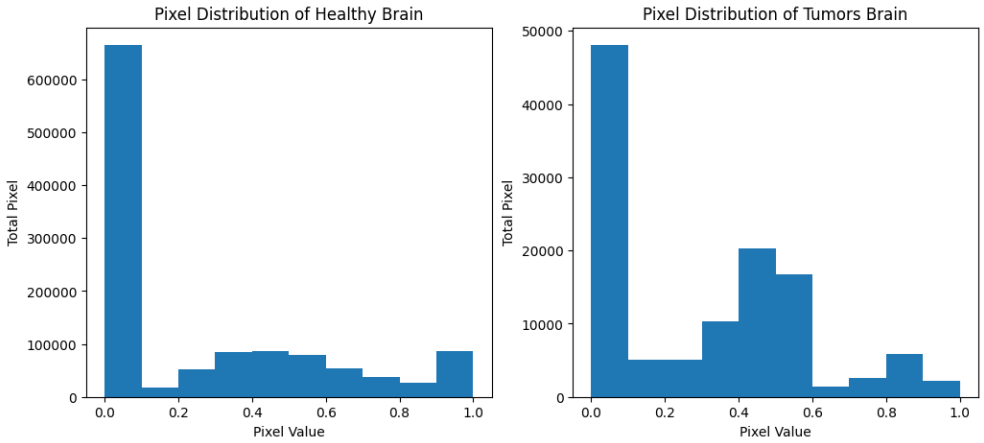
b. ***Data Preproccesing***

Berdasarkan explorasi yang telah dilakukan, data akan dibagi menjadi tiga *subset* yaitu *train set, validation set* dan *testing set.* Pembagian *subset* tersebut diperuntukan untuk melatih model dan menguji model. Rasio antar masing-masing *subset train : validation : testing* adalah 70 : 20 : 10. Sehingga dari 3000 gambar, masing masing dialokasikan 2100 gambar untuk *training set*, 600 gambar untuk *validation set* dan 300 gambar untuk *testing set.*



Gambar 4.2. Distribusi train - val - test set

Data yang telah dibagi menjadi tiga set akan dinormalisasi untuk mempermudah model mencapai konvergensi. Normalisasi dilakukan pada tiap gambar dengan membagi nilai tiap piksel dengan nilai maksimum pixel yaitu 255 yang menjadikan nilai tiap piksel berada pada rentang nilai 0 - 1.

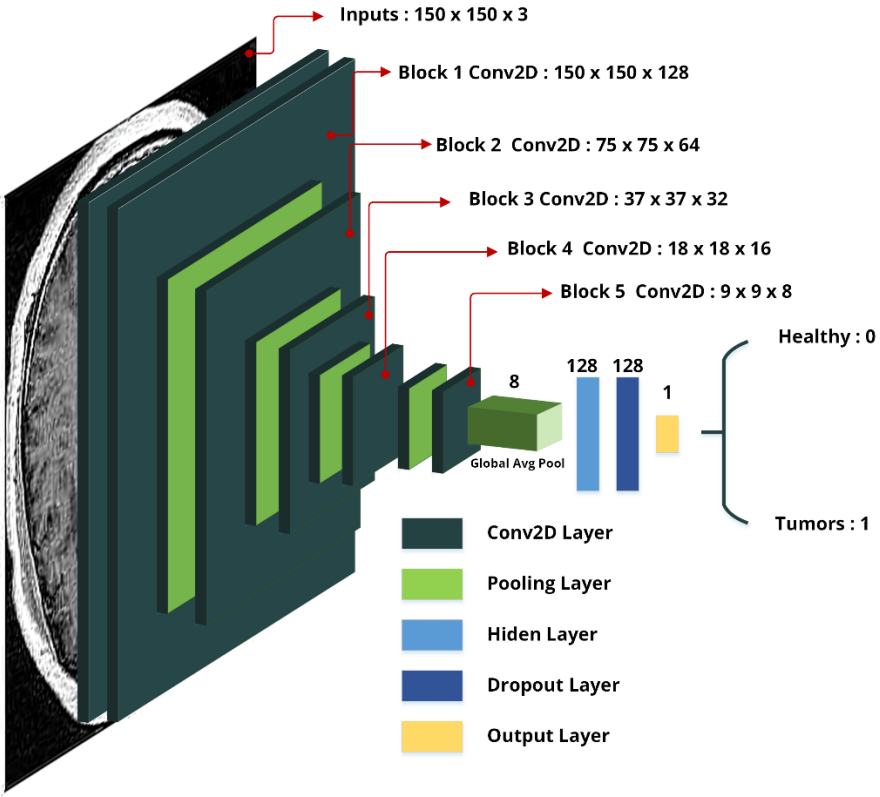


Gambar 4.3. Distribusi pixel gambar pindaian otak dengan MRI

c. ***Model Selection***

Model yang digunakan pada penelitian adalah *Deep Neural Network* yang terdiri atas *Convolutional layer, Max Pooling layer* dan *Dense (Neuron) layer.*

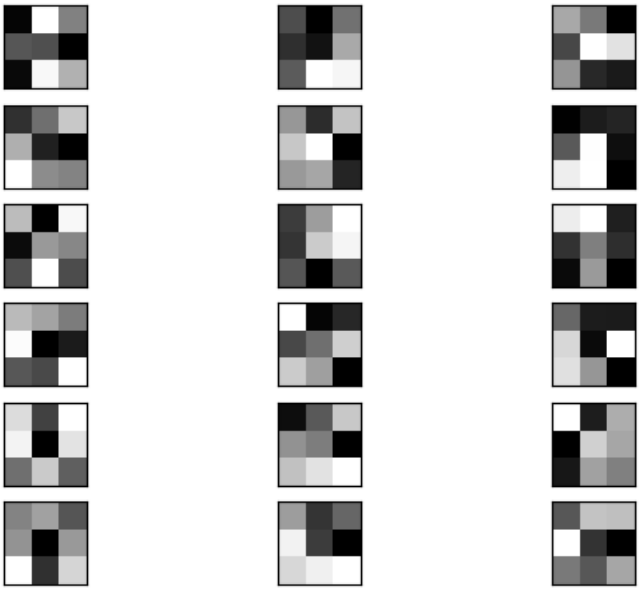
Layer-layer tersebut memiliki *hyperparameter* yang berfungsi sebagai nilai pengoperasian ditiap *layer.* Adapun model *plot* kombinasi layer dan *hyperparameter* yang digunakan sebagai berikut.



Gambar 4.4. Arsitektur Model

1. *Convolutionl Filter*

   Lapisan konvolusi beroperasi dengan mekanisme mangkalikan tiap filter konvolusi dengan gambar masukan menghasilkan n gambar, dimana n merupakan jumlah nilai filter yang diinisialisi pada hyperparameter.

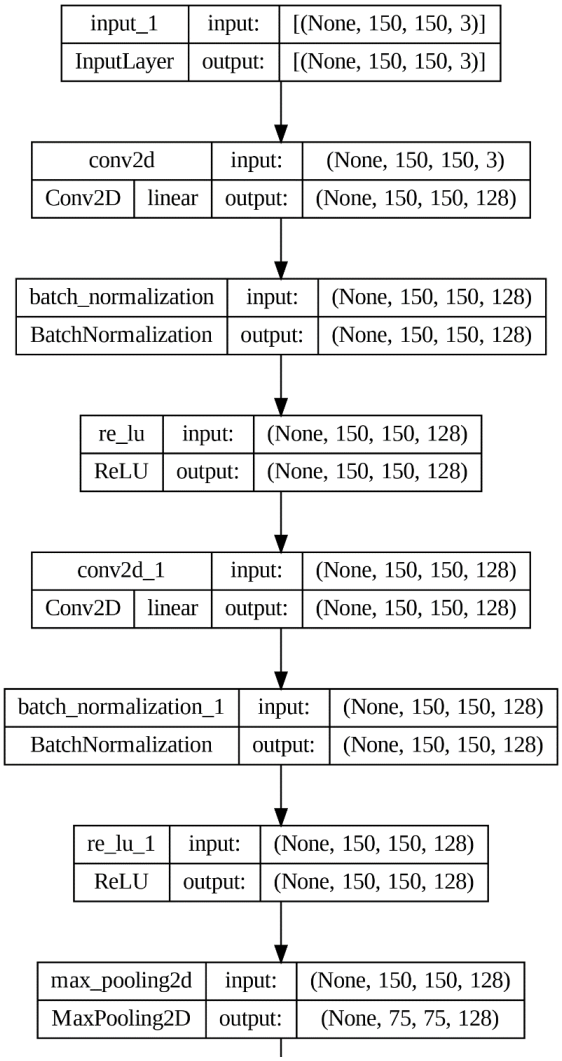


Gambar 4.5. Filter konvolusi

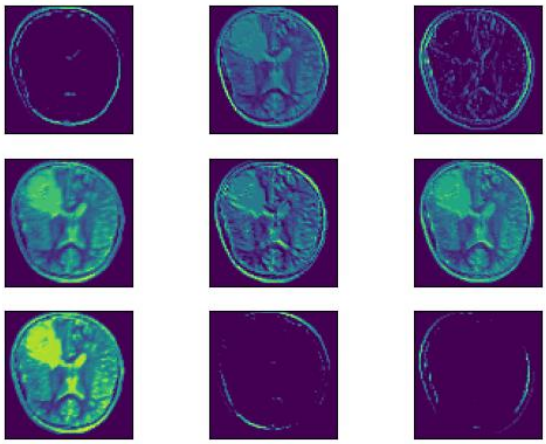
   Tiap gambar yang dihasilkan dari perkalian filter tersebut akan memberikan gambar baru dengan karakteristik yang berbeda, seperti gambar yang hanya menampilkan tepian gambar, latar belakang gambar maupun gambar objek pada gambar.

2. *First Convolutional Block Layer*

   Lapisan blok konvolusi pertama akan menerima input langsung dari gambar masukan dengan bentuk 150 x 150 x 3, dimana diakhir lapisan blok konvolusi pertama akan mereduksi ukuran gambar yang memiliki *feature* gambar yang tertangkap dan memberikan keluaran 75 x 75 x 128 gambar.



Gambar 4.6. Lapisan *Block Convolutional* pertama

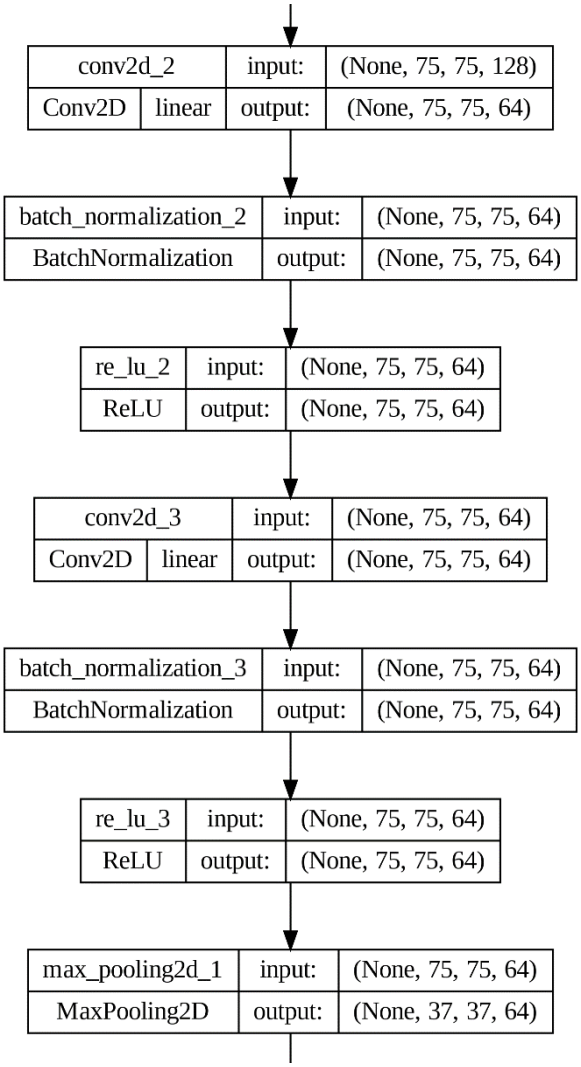


Gambar 4.7. Hasil *block convolutional* pertama

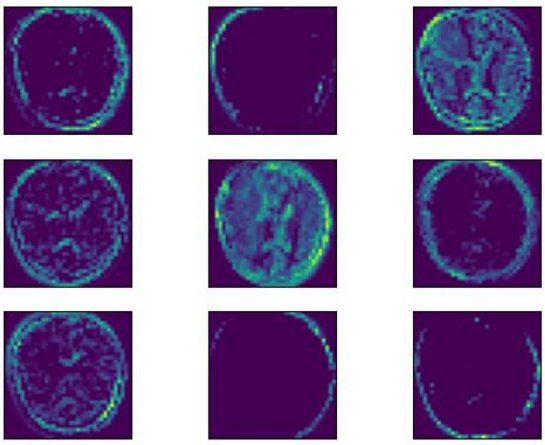
3. *Second Convolutional Block Layer*

   Pada lapisan kedua, *block convolutional* akan menerima keluaran dari lapisan sebelumnya untuk mendapatkan nilai *feature* gambar kembali dan mereduksinya untuk memperkecil ukuran gambar serta menyisakan bagian terpenting pada gambar.

   Block convolutional kedua menerima 75 x 75 x 128 masukan dan mengeluarkan 64 gambar baru dengan keluaran 37 x 37 x 64 yang sudah dikalikan dengan *filter convolutional*.



Gambar 4.8. Lapisan *block convolutional* kedua

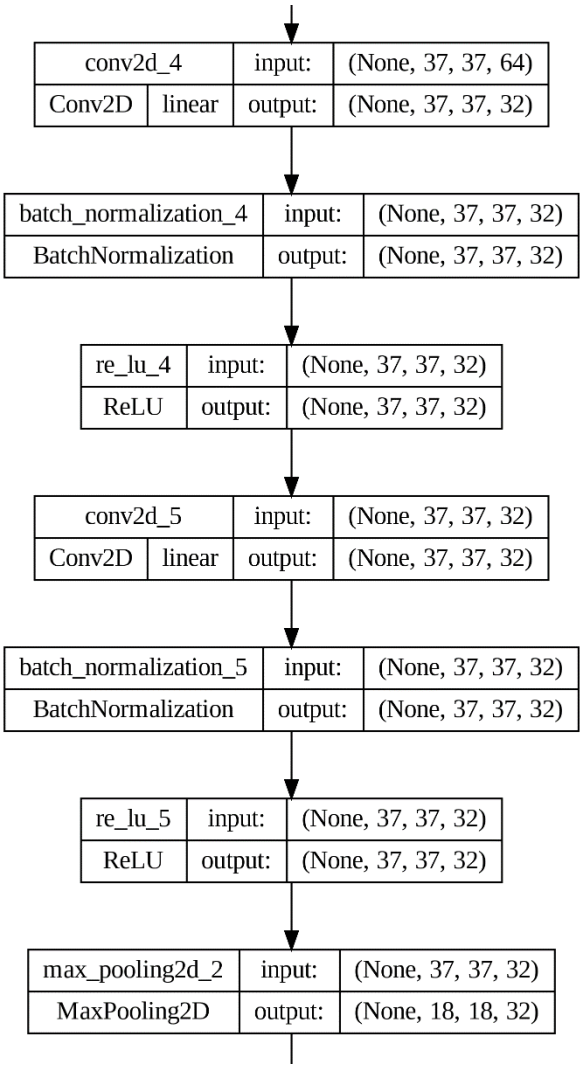


Gambar 4.9. Hasil konvolusi block kedua

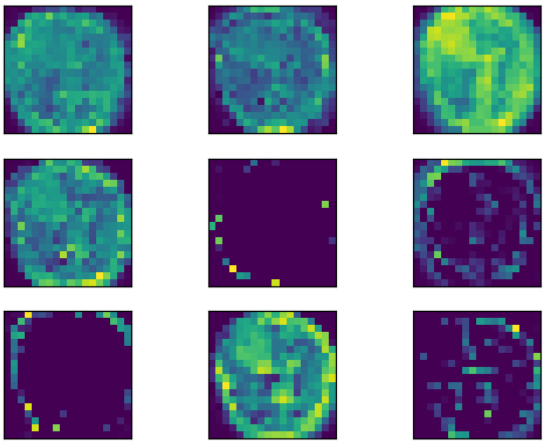
4. *Third Convolutional Block Layer*

Block convolutional ketiga kembali mengambil keluaran gambar dari lapisan sebelumnya sebanyak 37 x 37 x 64 atau 64 gambar baru dari satu gambar dengan ukuran 37 x 37 piksel.

Lapisan ketiga konvolusi menghasilkan keluaran berupa 32 gambar baru dengan ukuran 18 x 18 piksel. Pada lapisan ini gambar akan tampak *blur* namun memiliki *feature* yang berarti pada gambar masukan.



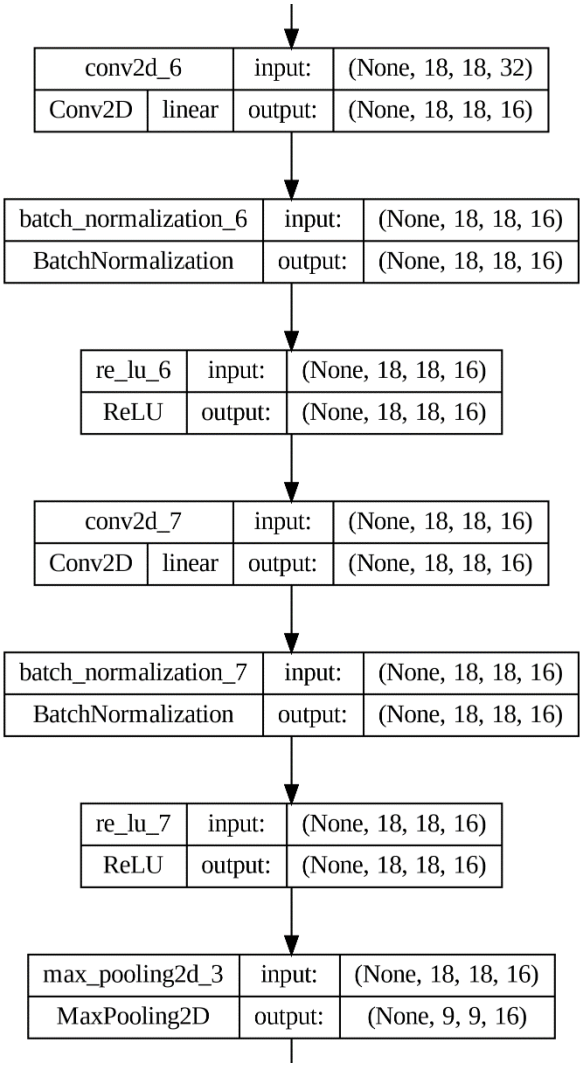
Gambar 4.10. Lapisan *block convolutional* ketiga



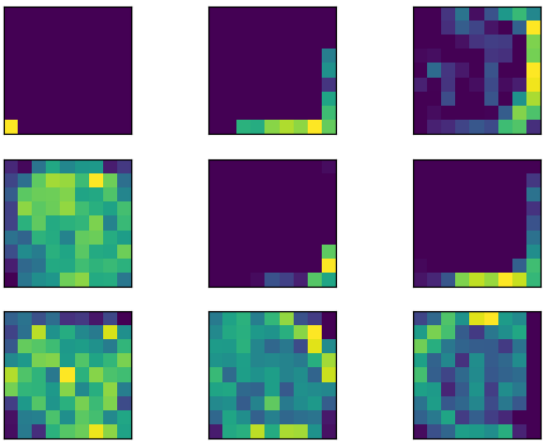
Gambar 4.11. Hasil konvolusi *block* ketiga

5. *Fourth Convolutional Block Layer*

   Lapisan konvolusi keempat akan menerima masukan berupa 32 gambar dengan ukuran 18 x 18 piksel dan menghasilkan 16 gambar berukuran 9 x 9 piksel. Pada lapisan ini gambar akan mulai tidak terlihat seperti gambar otak pada umumnya karena sudah diekstrak berulangkali pada *layer* sebelumnya.



Gambar 4.12. Lapisan *Block convolutonal* keempat

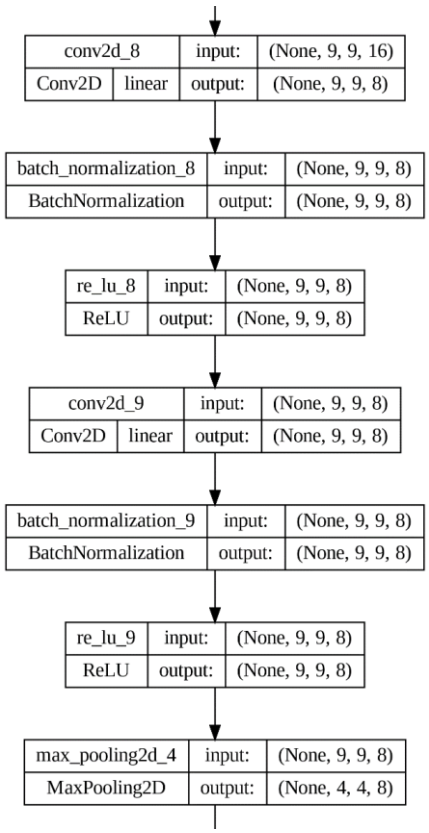


Gambar 4.13. Hasil konvolusi *block* keempat

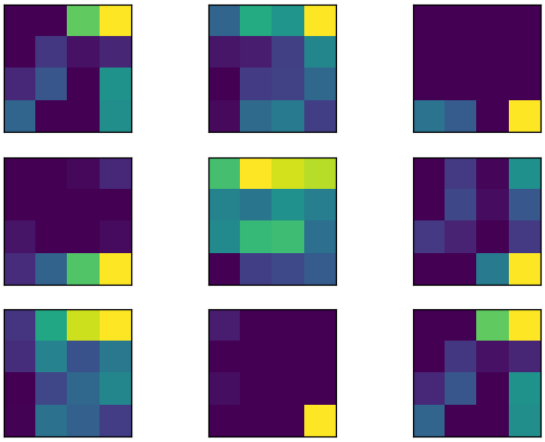
6. *Fifth Convolutional Block Layer*

Lapisan konvolusi kelima yang merupakan lapisan terakhir pada block convolutional akan memiliki feature unik pada gambar masukan yang berasal dari keluaran lapisan sebelumnya. Meskipun gambar yang dihasilkan tidak memiliki bentuk seperti otak, *feature* unik tersebutlah yang akan dimasukan kedalam lapisan neuron / dense untuk dihitung bobotnya pada tiap neuron hidden layer.

Lapisan ini akan menghasilkan 8 gambar berukuran 4 x 4 piksel. ukuran yang sangat kecil untuk dapat diinterpretasikan bagi mata manusia, namun kaya akan fitur unik pada gambar masukan.



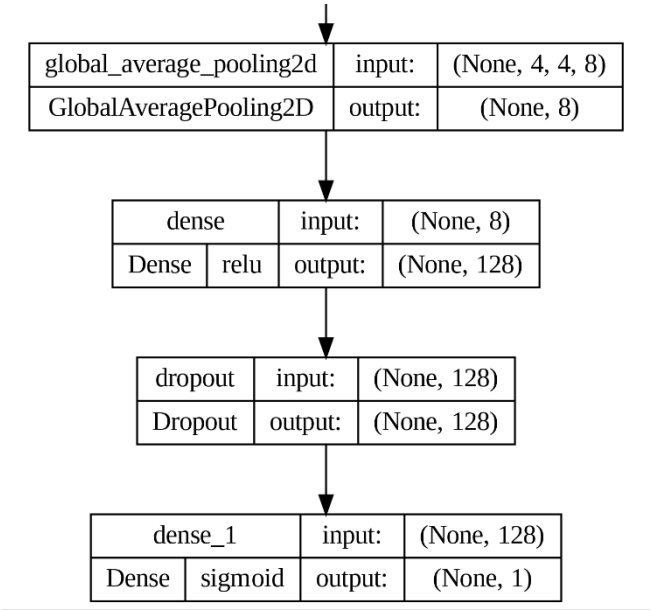
Gambar 4.14. Lapisan *block convolutional* kelima



Gambar 4.15. Hasil konvolusi *block* kelima

7. *Fully connected layer / dense layer*

   Lapisan terakhir pada model merupakan lapisan pembobotan jaringan syaraf tiruan. Sebelum keluaran dari lapisan terakhir konvolusi dimasukan ke dalam *neuron*, masukan akan di ubah dari *multidimetional array / multidimetional tensor* ke bentuk *one dimensional array / tensor* melalui lapisan *global average pooling*. Lalu setelah itu masukan dapat dihitung bobotnya pada tiap *neuron* atau *perceptron* dengan lebih mudah dan mencapai nilai konvergensi dan memprediksi nya di layer *output*.

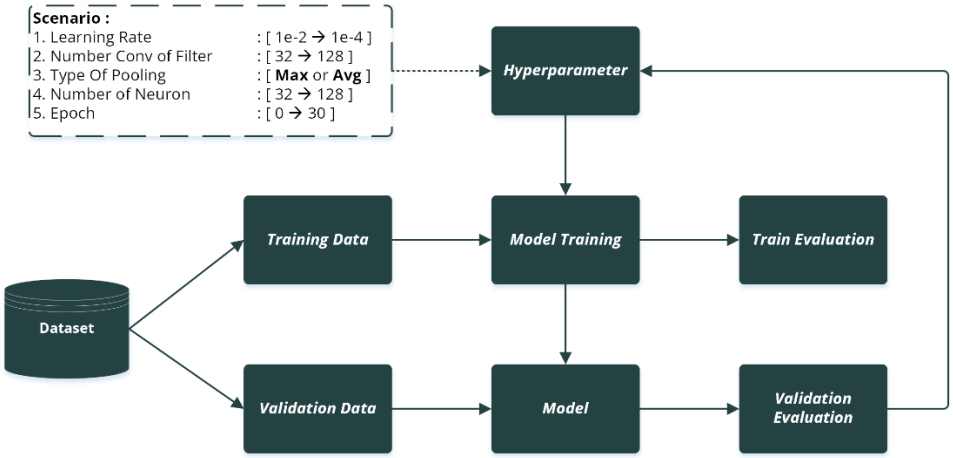


Gambar 4.16. Lapisan akhir jaringan syaraf tiruan

Lapisan *dropout* yang terletak sebelum lapisan *output* berfungsi sebagai *regulerization* dimana *output* dari *dense layer* sebelumnya akan ditiadakan sebanyak 20 %. Hal ini bertujuan untuk mengurangi keluaran yang bernilai sama dan menghindari *overfiting* pada model.

d. ***Hyperparameter Tuning***

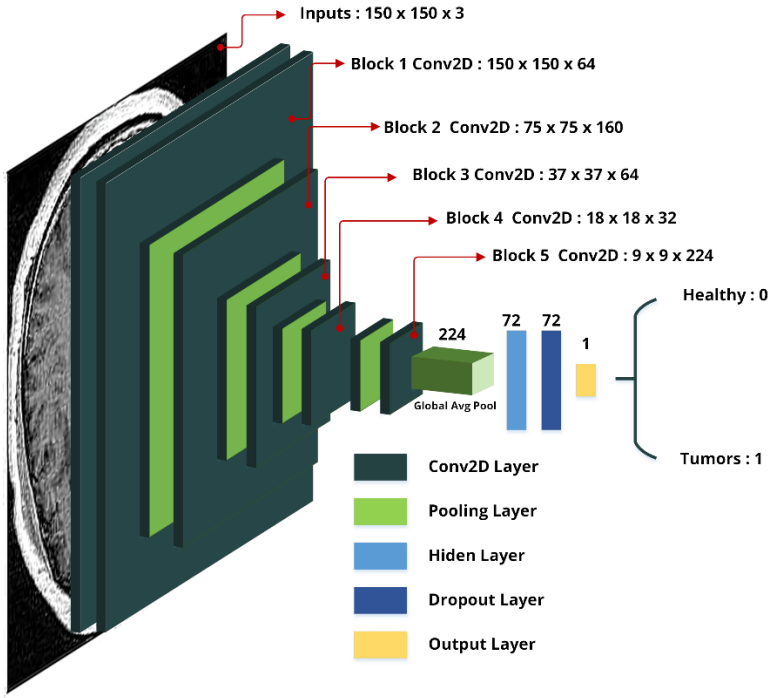
Pemodelan *deep learning* seringkali mendapatkan model yang *overfiting , underfiting* atau bahkan model yang tidak memiliki *hyperparameter* optimal untuk mendapatkan nilai maksimal. Masalah dalam menentukan model yang optimal dengan hasil yang maksimal dapat memakan waktu yang lama dan membuang sumber daya bila tujuannya tidak tercapai. Oleh karena itu, metode *hyperparameter tuning* dinilai dapat mengatasi permasalahan tersebut.



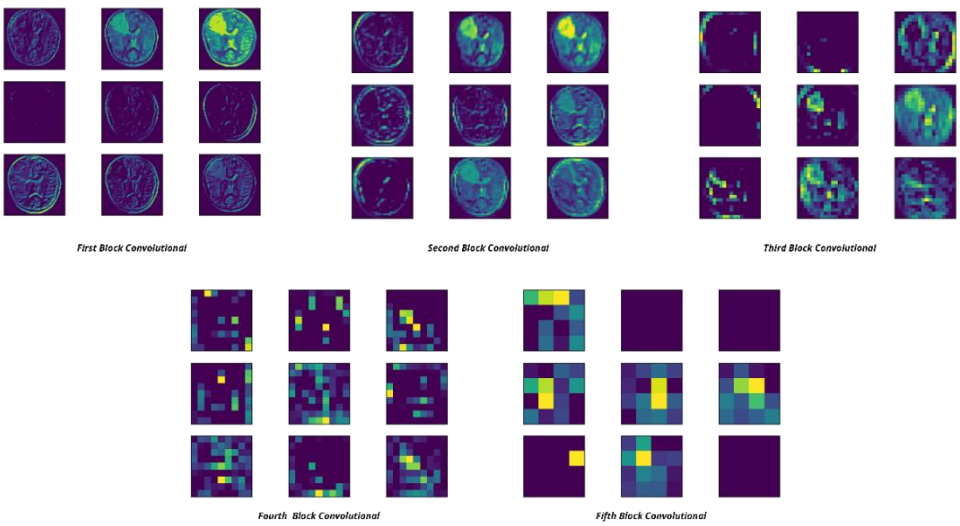
Gambar 4.17. Skenario *hyperparameter tuning*

Metode *hyperparameter tuning* bekerja dengan mekanisme menguji tiap skenario yang telah diinisialisasi dan menjalankan nya dengan iterasi yang relatif pendek, lalu memonitor perubahan nilai nya. Bila tidak terdapat perubahan yang signifikan pada model pada rentang iterasi yang pendek, model akan diuji dengan nilai hyperparameter yang berbeda.

Penerapan *hyperparameter tuning* pada model yang diusulkan menghasilkan model dengan nilai optimal dan nilai maksimal yang dapat dilihat pada gambar 4.18 dan 4.19 serta *learning rate* terbaik yaitu (50178 x 10^{-8}).



Gambar 4.18. Arsitektur hypermodel



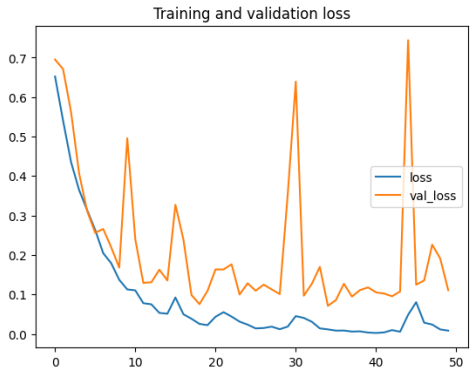
Gambar 4.19. Hasil Konvolusi Hypermodel

Berdasarkan gambar 4.19, hasil konvolusi dapat mengekstrak dan mensegmentasi bagian tumor dengan lebih akurat dibandingkan model sebelumnya. Memberikan *feature* yang lebih berarti dan kaya untuk model yang dijalankan.

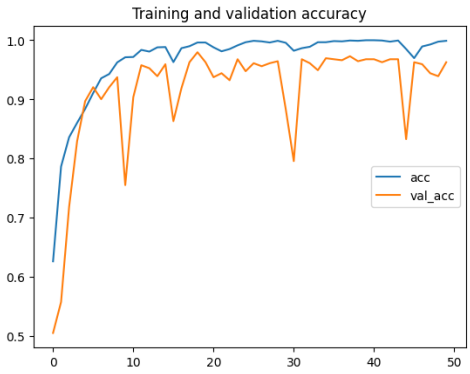
e. ***Evaluation***

Model yang telah dilatih akan dievalusi berdasarkan hasil prediksinya terhadap data yang belum pernah dilihat pada *set testing.* Hasil prediksi tersebut dinilai berdasarkan metrik *loss, accuracy, sensitivity, specificity, precision, F1-Score dan dice similarity coeficient (DSC).*

Pada proses pelatihan model dengan 50 kali epoch / iterasi didapatkan akurasi sebesar 96,6 % dengan *loss* sebesar 12 %. Meskipun mendapatkan akurasi yang relatif tinggi, grafik yang cenderung fluktuatif menandakan model kesulitan dalam mencapai nilai konvergen atau mengenali pola pada gambar.

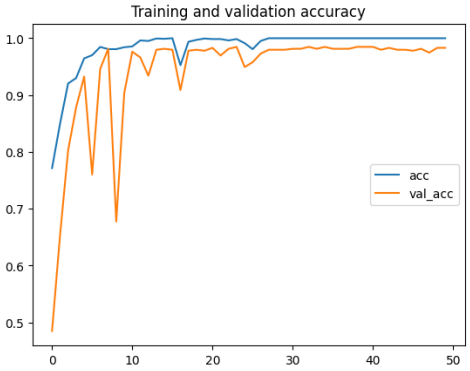


Gambar 4.20. Kinerja pelatihan terhadap nilai *loss model*

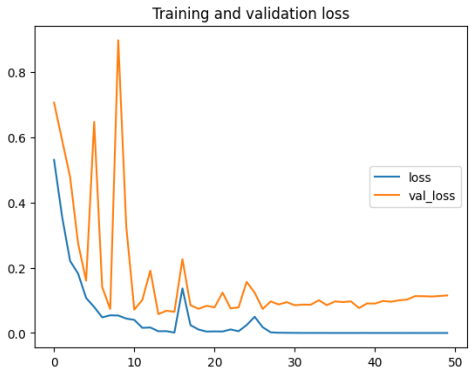


Gambar 4.21. Kinerja pelatihan terhadap nilai akurasi model

Sedangkan proses pelatihan pada *hyper model* dengan 50 kali epoch / iterasi didapatkan akurasi sebesar 98,6 % dengan nilai loss sebesar 5,8 %. Bila dilihat pada gambar 4.22 dan 4.23 grafik melaindai yang menunjukan konvergensi setelah melewati lebih dari 30 kali epoch dan menandakan model berhasil mengenali pola pada gambar lebih baik dibandingkan model sebelumnya serta menandakan *hyperparameter tuning* berhasil meningkatkan performa model dengan nilai *hyperparameter* yang paling optimal dengan hasil yang maksimal.

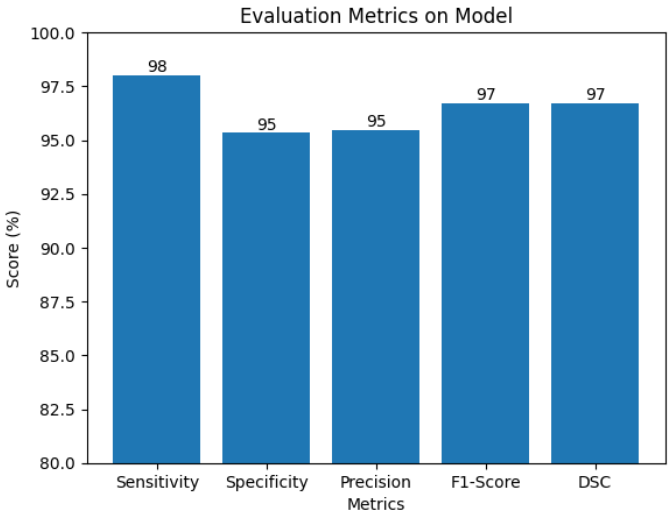


Gambar 4.22. Kinerja pelatihan terhadap nilai akurasi hyper model



Gambar 4.23. Kinerja pelatihan terhadap nilai *loss* hyper model

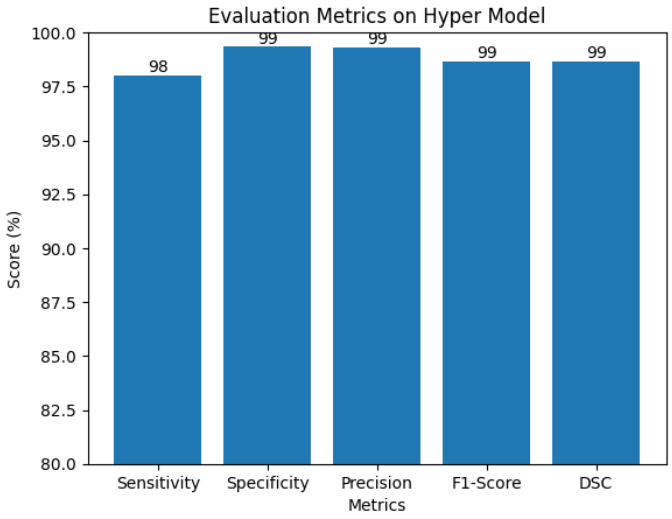
Adapun perfoma model menggunakan metrik *sensitivity*, *specificity*, *precision*, *F1-Score dan dice similarity coeficient* memberikan performa yang cukup baik dengan hasil diatas 90 %.



Gambar 4.24. Performa metrik model

Performa *hypermodel* pada metrik *sensitivity*, *specificity*, *precision*, *F1-Score dan dice similarity coeficient* menunjukan nilai diatas performa model sebelumnya, dimana masing-masing metrik memiliki nilai sebesar

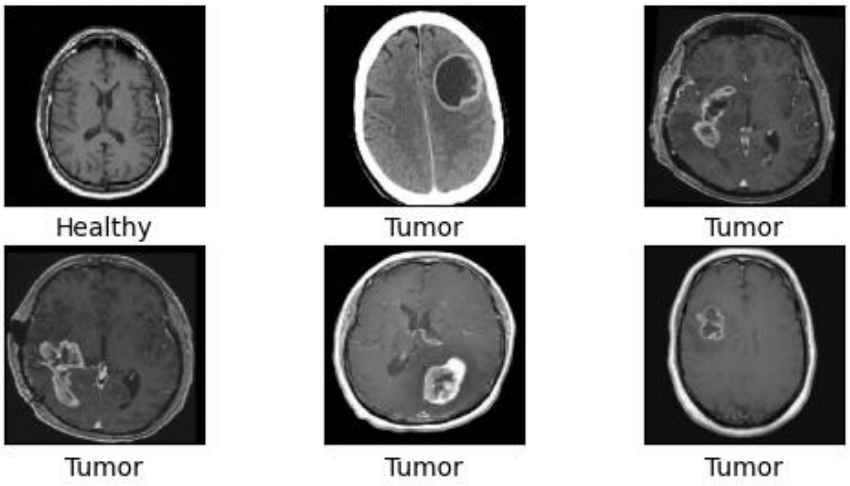
98%, 99,3%, 99,3%, 98,6% dan 98,6%. Performa yang meningkat tersebut menunjukan bahwa hypermodel memiliki lapisan dengan *hyperparameter* paling optimal untuk kasus deteksi tumor otak pada dataset yang digunakan.



Gambar 4.25. Performa metrik *hypermodel*

f. ***Testing / Deployment***

Berdasarkan hasil performa model dan *hypermodel* pada metrik *sensitivity*, *specificity*, *precision*, *F1-Score* dan *dice similarity coeficient*, *hypermodel* memiliki nilai tertinggi sehingga *hypermodel* dipilih sebagai model yang digunakan untuk proses ujicoba prediksi tumor otak pada gambar pindaian *magnetik resonance imaging*. *Hypermodel* akan diujicoba langsung melalui *platform google colaboratory* untuk memprediksi apakah gambar masukan terdeteksi sebagai otak terdiagnosis tumor atau tidak.



Gambar 4.26. Hasil prediksi *hyper model* pada gambar MRI

4.2 Pembahasan

Prediksi tumor otak pada gambar resonansi magnetik dengan metode *deep neural network* memberikan performa yang cukup baik dalam menentukan apakah otak terdiagnosis tumor atau dalam keadaan sehat. Hal ini didasarkan pada nilai metrik *accuracy, sensitivity, specificity, precision, F1-score,* dan *dice similarity coeficient* yang mencapai performa diatas 95%, dimana masing-masing metrik memiliki nilai 98% *accuracy*, 98% *sensitivity*, 99,3% *specificity*, 99,3% *precision*, 98,6% *F1-score* dan 98,6% *dice similarity coeficient.*

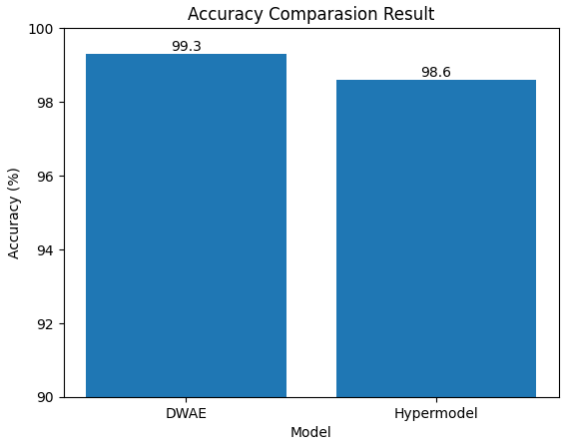
Beberapa metrik tersebut bahkan telah melebihi perfoma model oleh Isselmou Abd El Kader, Guizhi Xu, Zhang Shuai, Sani Saminu, Imran Javid, Isah Salim Ahmad pada penelitiannya yang menggunakan metode *deep wavelet auto encoder* dengan kasus serupa, yaitu prediksi tumor otak pada gambar MRI [5]. Adapun interpretasi setiap metrik yang digunakan sebagai evaluasi pada model *deep learning* yang digunakan untuk memprediksi tumor dan komparasi

nya terhadap penelitian [5] adalah sebagai berikut.

a. ***Accuracy***

Metrik *accuracy* menentukan performa model dalam memprediksi sample yang terprediksi secara benar dari total sampel yang diujicoba ke dalam model. Artinya semakin banyak prediksi yang benar pada sampel yang diujicoba maka nilai *accuracy* model semakin tinggi.

Nilai *accuracy* yang didapatkan oleh *hypermodel* mencapai 98,6 % pada *test set*. Total keseluruhan *test set* digunakan adalah sebesar 300 sampel yang berarti dari 300 sample yang termasuk didalamnya sampel tumor dan healthy, 294 sampel terprediksi / terklasifikasi oleh model dengan benar dan 6 diantaranya terprediksi salah. Bila dibandingkan dengan penelitian [5] performa *accuracy* dapat dilihat pada gambar 4.27.



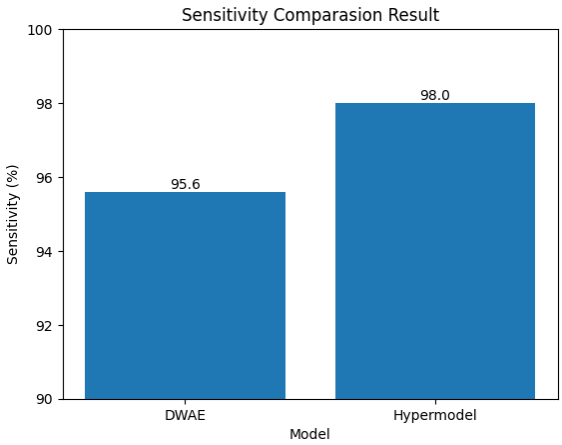
Gambar 4.27. Komparasi *accuracy* pada penelitian terkait dengan *hypermodel*

b. ***Sensitivity***

Metric *sensitivity* menentukan performa model dalam memprediksi sample yang hanya terdiagnosis tumor otak dan terprediksi benar. Artinya semakin

banyak sampel terprediksi benar dari seluruh sampel yang hanya terdiagnosisi tumor maka nilai *sensitivity* semakin tinggi.

Nilai *sensitivity* yang didapatkan oleh *hypermodel* mencapai 98 % pada *test set tumors.* Dari 150 sampel terdiagnosis otak, *hypermodel* berhasil memprediksi secara benar sebanyak 147 sampel dan 3 diantaranya terprediksi salah. Bila dibandingkan dengan penelitian [5] yang hanya mencapai nilai performa *sensitivity* 95,6%, *hypermodel* memiliki nilai yang relatif lebih tinggi.



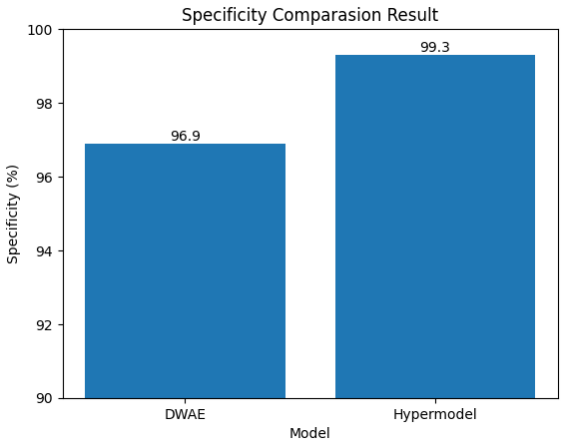
Gambar 4.28. Komparasi *sensitivity* pada penelitian terkait dengan hypermodel

c. ***Specificity***

Metric *specificity* menentukan performa model dalam memprediksi sample yang hanya sehat dan terprediksi benar. Artinya semakin banyak sampel terprediksi benar dari seluruh sampel yang hanya sehat maka nilai sensitivity semakin tinggi.

Nilai *specificity* yang didapatkan oleh *hypermodel* mencapai 99,3 %

pada *test set* tumors. Dari 150 sampel terdiagnosis otak, *hypermodel* berhasil memprediksi secara benar sebanyak 149 sampel dan 1 diantaranya terprediksi salah. Bila dibandingkan dengan penelitian [5] yang hanya mencapai nilai performa *specificity* 95,6%, *hypermodel* memiliki nilai yang relatif lebih tinggi.



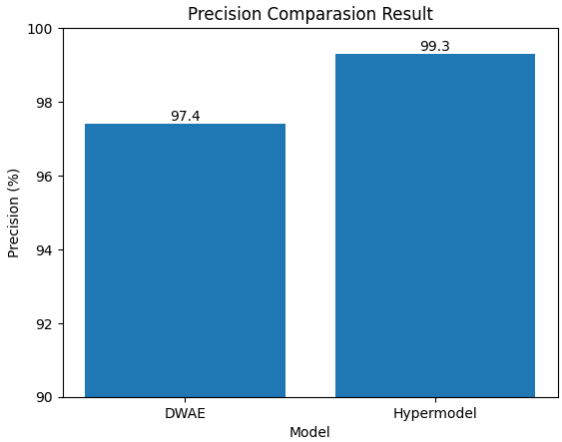
Gambar 4.29. Komparasi *specificity* pada penelitian terkait dengan hypermodel.

d. ***Precision***

Metric *precision* menentukan performa model dalam memprediksi sampel yang terprediksi sebagai tumor dengan benar dari keseluruhan sampel terprediksi tumor. Artinya semakin banyak sampel terprediksi tumor dengan benar dari keseluruhan sample yang terprediksi tumor maka *precision* semakin tinggi.

Precision yang didapatkan oleh *hypermodel* mencapai 99,3% pada *test set*. Artinya dari 150 sampel yang terprediksi tumor, model berhasil memprediksi dengan benar tumor sebanyak 149 sampel dan 1 sampel

terprediksi tumor secara salah. Bila dibandingkan dengan penelitian [5] yang hanya mencapai nilai performa *precision* 97,4%, *hypermodel* memiliki nilai yang relatif lebih tinggi.



Gambar 4.30. Komparasi *precison* pada penelitian terkait dengan hypermodel

e. ***F1-Score***

Metrik *F1-score* menentukan performa model dalam keseimbangan terprediksi secara benar positif dan benar negatif tumor. Artinya semakin banyak sample yang terprediksi benar positif tumor dan benar negatif maka nilai *F1-score* semakin tinggi (mencapai keseimbangan).

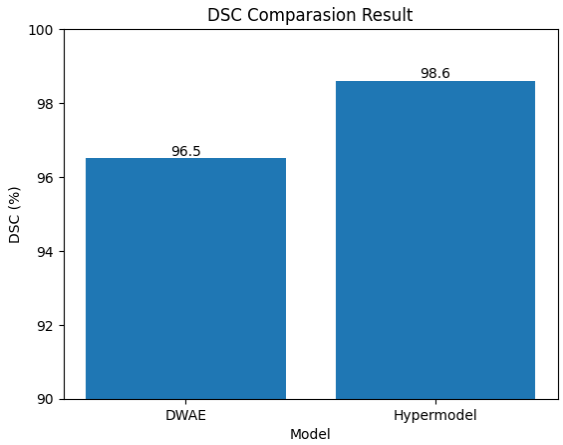
F1-score yang didapatkan oleh *hypermodel* adalah sebesar 0.986 dimana nilai maksimal nya 1. Dari 300 sampel *test set* model mencapai nilai keseimbangan 0.986 dalam memprediksi tumor dengan benar dan benar negatif.

f. ***Dice Similarity Coeficient***

Metrik *dice similarity coeficient* menentukan performa model dengan membandingkan hasil prediksi otak terdiagnosis tumor dan non-tumor pada

nilai aktual sampel terdiagnosis tumor dan non-tumor. Artinya semakin nilai prediksi mendekati nilai aktual sampel, nilai metrik akan semakin tinggi.

Dice similarity coeficient yang didapatkan oleh *hypermodel* adalah sebesar 98, 6%. Dari 300 sampel aktual *test set*, nilai prediksi model mendekati nilai aktual sebanyak 296 sampel. Bila dibandingkan dengan penelitian [5] yang memiliki performa *dice similarity* 96,5%, *hypermodel* mencapai nilai kemiripan lebih tinggi.



Gambar 4.31. Komparasi DSC pada penelitian terkait dengan hypermodel