

Optimasi *Variational Quantum Classifier* (VQC) Pada Klasifikasi Gambar Perkebunan Kelapa Sawit

Desi Amirullah ¹⁾, Lipantri Mashur Gultom ²⁾

^{1,2}Teknik Informatika, Politeknik Negeri Bengkalis

E-mail: andes@polbeng.ac.id, lipantri@polbeng.ac.id

Abstract

This study explores the optimization of the *Variational Quantum Classifier* (VQC) for classifying plantation images, particularly distinguishing between oil palm and non-oil palm trees. Previous quantum models such as Quantum Convolutional Neural Networks (QCNN) demonstrated high accuracy but suffered from high complexity and *qubit* requirements. To address this limitation, this research employs a more modular and efficient quantum model, the VQC, optimized through the exploration of 18 combinations of feature maps (PauliFeatureMap, ZFeatureMap, ZZFeatureMap), *ansatz* circuits (RealAmplitudes, TwoLocal, EfficientSU2), and optimizers (COBYLA, SPSA). The dataset comprises 220 images resized to 32×32 *pixels* and preprocessed through standard scaling, PCA (to 4 dimensions), and normalization to [-1,1]. The best performance was achieved using the combination of ZZFeatureMap, EfficientSU2, and COBYLA, yielding an accuracy of 93%. Evaluation metrics include accuracy, precision, recall, F1-score, and specificity. The results reveal that the selection and interaction of feature map, *ansatz*, and optimizer significantly affect classification performance. These findings suggest that VQC is a promising alternative for image classification in digital agriculture, offering high accuracy with lower quantum resource demands.

Keywords: *Variational Quantum Classifier, Oil Palm Image Classification, Quantum Machine Learning, Feature Map, Quantum Optimizer*

PENDAHULUAN

Sektor perkebunan kelapa sawit memainkan peran vital dalam perekonomian global dan klasifikasi spesies pohon yang akurat sangat krusial untuk meningkatkan produktivitas serta efisiensi panen. Namun, klasifikasi citra konvensional sering kali memerlukan sumber daya komputasi yang besar. Seiring dengan kemajuan pesat dalam komputasi kuantum, model pembelajaran mesin kuantum (*Quantum Machine Learning*) seperti *Variational Quantum Classifier* (VQC) menawarkan alternatif yang menjanjikan. VQC mengintegrasikan prinsip-prinsip kuantum, berpotensi memberikan keunggulan komputasi dalam pemrosesan data. Saat ini teknologi kuantum telah diterapkan dalam bidang pengolahan citra pertanian, salah satunya melalui model *Quantum Convolutional Neural Network* (QCNN) seperti yang ditunjukkan dalam penelitian (Amirullah, 2025). Model QCNN menunjukkan potensi yang menjanjikan dalam klasifikasi pohon kelapa sawit dengan akurasi tinggi namun masih

menyisakan beberapa kelemahan dan ruang pengembangan, terutama dalam konteks pengolahan data terbatas dan kompleksitas model kuantum. penelitian sebelumnya secara spesifik hanya berfokus pada QCNN yang merupakan salah satu jenis model pembelajaran kuantum hibrida.

Meskipun QCNN terbukti efektif, arsitekturnya yang melibatkan lapisan konvolusi dan *pooling* kuantum memiliki kompleksitas yang tinggi dan membutuhkan banyak *qubit* yang mungkin tidak efisien untuk semua jenis masalah (Alami, 2024) (Bagoun, 2024). Penelitian ini juga tidak secara eksplisit mengeksplorasi model kuantum hibrida yang lebih sederhana dan fleksibel seperti *Variational Quantum Classifier* (VQC). VQC memiliki arsitektur yang lebih modular dan mudah disesuaikan, berpotensi untuk memberikan performa yang sebanding atau bahkan lebih baik untuk pengklasifikasian yang sama namun dengan tuntutan komputasi kuantum yang berpotensi lebih rendah dan implementasi yang lebih sederhana (Adermann, 2025) (Fathi, 2024). VQC memiliki potensi untuk menyelesaikan masalah klasifikasi kompleks dengan efisiensi tinggi khususnya pada data *non-linier* yang sulit dipisahkan oleh model konvensional (Maheshwari, 2021) (Laskaris, 2023).

Namun demikian, penerapan VQC dalam domain pertanian digital masih belum banyak dieksplorasi, khususnya untuk klasifikasi citra tanaman tropis seperti kelapa sawit. Penelitian dalam bidang ini menjadi penting untuk menjawab pertanyaan apakah algoritma kuantum seperti VQC benar-benar mampu memberikan keunggulan dalam konteks praktis seperti klasifikasi citra lahan perkebunan. Oleh karena itu, terdapat celah penelitian untuk mengeksplorasi apakah VQC dapat menjadi alternatif yang lebih efisien dan *scalable* dibandingkan QCNN untuk klasifikasi gambar kelapa sawit.

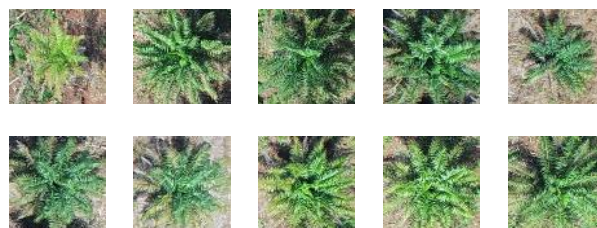
Berdasarkan latar belakang tersebut, rumusan masalah dalam penelitian ini adalah bagaimana merancang arsitektur *Variational Quantum Classifier* (VQC) yang sesuai untuk klasifikasi gambar perkebunan kelapa sawit ?, bagaimana optimasi arsitektur VQC seperti pemilihan *ansatz* dan metode *encoding* data yang sesuai dapat mempengaruhi akurasi klasifikasi gambar ? dan apa saja metrik evaluasi yang paling tepat digunakan untuk mengukur keberhasilan model VQC dalam klasifikasi citra kelapa sawit ?

Penelitian sebelumnya telah menunjukkan potensi VQC dalam berbagai tugas klasifikasi (Adermann, 2025) (Khan, 2025) (Laskaris, 2023) (Mattern, 2021) (Parthasarathi, 2025) (Sathan, 2024) (Souza, 2025). Namun, performa model ini sangat bergantung pada pemilihan arsitektur sirkuit kuantumnya, termasuk FeatureMap yang berfungsi untuk meng-encoding data, Ansatz sebagai sirkuit yang dapat dilatih, dan Optimizer yang menyesuaikan parameter

model. Oleh karena itu, penelitian ini bertujuan untuk secara sistematis mengeksplorasi dan mengevaluasi berbagai kombinasi parameter VQC untuk mengidentifikasi konfigurasi optimal yang secara efektif mengklasifikasi citra perkebunan kelapa sawit.

METODE PENELITIAN

Dalam penelitian ini menggunakan dataset yang berisikan 220 gambar digital berukuran 32×32 pixel yang terdiri dari 110 gambar pohon kelapa sawit dan 110 gambar bukan pohon kelapa sawit. Berikut ini sample gambarnya:



Gambar 1. Sampel gambar pohon sawit



Gambar 2. Sampel gambar bukan pohon sawit

Dataset tersebut dibagi menjadi dua bagian yaitu data pelatihan sebanyak 80% (176 gambar) dan data pengujian 20% (44 gambar).

Tahapan *preprocessing* terdiri dari 3 tahap yang secara berurutan mulai dari *Standar Scaler* berfungsi untuk standarisasi data, yaitu mengubah distribusi data sehingga memiliki rata-rata (μ) 0 dan standar deviasi (σ) 1. Hal ini dilakukan dengan menggeser titik pusat data ke nol dan menyesuaikan skala variansnya. Setiap *pixel* dalam gambar dapat dianggap sebagai satu fitur. Dalam dataset, gambar-gambar mungkin memiliki kondisi pencahayaan yang berbeda-beda. Misalnya, beberapa gambar lebih gelap dengan nilai *pixel* rendah, sementara yang lain lebih terang dengan nilai *pixel* tinggi. Variasi nilai ini bisa membuat model bias, di mana model lebih terpengaruh oleh gambar yang lebih terang karena memiliki nilai *pixel* yang lebih besar. Proses standarisasi nilai *pixel* dari semua gambar sehingga kondisi pencahayaan tidak lagi menjadi faktor dominan. Setelah diproses, nilai *pixel* di semua

gambar akan memiliki bobot yang setara, memungkinkan model untuk fokus pada fitur-fitur yang lebih substansial seperti bentuk dan tekstur pohon kelapa sawit.

Kemudian PCA (*Principal Component Analysis*) digunakan untuk mengurangi jumlah fitur dalam data sambil mempertahankan informasi paling penting. PCA mencari arah (disebut *principal components*) di mana variasi data paling besar. Gambar 32×32 *pixel* memiliki $32 \times 32 = 1024$ fitur (satu fitur per *pixel*). Saat ini, perangkat keras kuantum (era NISQ) sangat terbatas dalam jumlah *qubit* yang bisa digunakan. Meng-*encoding* 1024 fitur membutuhkan 1024 *qubit* yang tidak mungkin dilakukan. Penyelesaian masalah ini dengan mereduksi data dari 1024 dimensi menjadi hanya 4 dimensi (4 *qubit*). Ini berarti PCA menemukan empat kombinasi fitur yang paling penting untuk membedakan gambar "Kelapa Sawit" dari "Bukan Kelapa Sawit" dan membuang informasi yang tidak relevan. Proses ini sangat krusial untuk membuat model kuantum yang layak secara komputasi dan dapat diimplementasikan dengan jumlah *qubit* yang tersedia.

Tahapan *preprocessing* yang terakhir adalah *MinMaxScaler* digunakan untuk Normalisasi data ke dalam rentang spesifik dalam hal ini (-1, 1). Setelah data direduksi dimensinya oleh PCA, nilai-nilai fitur yang baru mungkin masih berada dalam rentang yang tidak cocok untuk sirkuit kuantum. Beberapa sirkuit kuantum *feature map* (seperti *ZZFeatureMap* dan *ZFeatureMap*) memerlukan nilai input dalam rentang [-1, 1].

Selanjutnya, dalam penelitian ini menggunakan tiga kombinasi *feature map* yang berfungsi sebagai "penerjemah" dalam perubahan data gambar klasik menjadi data kuantum sehingga pola-pola data dapat diproses oleh sirkuit kuantum (Souza, 2025) (Sathan, 2024) (Mattern, 2021). Setelah gambar diproses oleh PCA, setiap gambar diwakili oleh 4 nilai numerik (4 dimensi). Fungsi dari *feature map* adalah mengambil 4 nilai ini dan menggunakannya untuk memanipulasi keadaan *qubit* atau bisa dibilang "meng-*encoding*" data. Pertama ialah *PauliFeatureMap*, menggunakan gerbang-gerbang Pauli (X, Y, Z) namun dalam kombinasi yang lebih kompleks. Gerbang ini biasanya melibatkan rotasi yang bergantung pada nilai data dan meng-*encoding* setiap fitur gambar. Perbedaannya terletak pada basis yang digunakan untuk *encoding*. Meskipun mungkin lebih kuat dari *ZFeatureMap*, *PauliFeatureMap* tetap fokus pada properti individual dari setiap fitur bukan pada interaksi antar fitur. Kedua ialah *ZFeatureMap*, menggunakan gerbang rotasi *qubit* di sumbu Z (R_z). Sudut rotasi untuk setiap *qubit* diatur sesuai dengan nilai dari fitur data gambar. Jadi, jika data memiliki 4 dimensi (x_1, x_2, x_3, x_4), *ZFeatureMap* akan meng-*encoding* data tersebut dengan melakukan rotasi sebesar $R_z(x_1)$ pada *qubit* pertama, $R_z(x_2)$ pada *qubit* kedua dan

seterusnya. *ZFeatureMap* secara efektif meng-*encoding* setiap fitur gambar secara independen. Metode ini secara efektif melakukan *encoding* setiap fitur gambar secara independen, memberikan nilai individu dari setiap fitur kepada model tanpa secara eksplisit menyajikan informasi tentang bagaimana fitur-fitur tersebut saling berhubungan atau berinteraksi. Ketiga ialah *ZZFeatureMap* tidak hanya menggunakan gerbang rotasi di sumbu Z (R_Z) untuk meng-*encoding* fitur individual tetapi juga menggunakan gerbang *entangling* seperti CNOT atau CZ. Gerbang *entangling* ini berfungsi untuk menghubungkan atau "mengikat" dua *qubit* atau lebih. Rotasi gerbang ini bergantung pada interaksi antara dua atau lebih fitur data (misalnya x_1 dan x_2). *ZZFeatureMap* tidak hanya melihat setiap fitur gambar secara terpisah tetapi juga secara eksplisit meng-*encoding* interaksi dan korelasi antar fitur. Dalam konteks gambar, ini berarti sirkuit kuantum dapat "mempelajari" hubungan antara *pixel* yang berbeda atau area yang berbeda dalam gambar. Kemampuan ini sangat penting untuk klasifikasi gambar, di mana pola-pola seperti bentuk atau tekstur seringkali berasal dari hubungan kompleks antar *pixel*, bukan dari nilai satu *pixel* saja. Inilah alasan mengapa *ZZFeatureMap* seringkali memberikan performa terbaik dalam penelitian ini.

Berikutnya penggunaan *Ansatz*, dalam konteks *Variational Quantum Classifier* (VQC), *Ansatz* adalah inti dari model yang dapat dilatih. Jika *Feature map* berfungsi untuk menerjemahkan data gambar ke dalam ruang kuantum maka *Ansatz* adalah sirkuit kuantum yang melakukan komputasi dan mempelajari pola-pola dari data tersebut (Parthasarathi, 2025). Fungsi utamanya adalah mencari dan menyimpan informasi yang membedakan satu kelas dari kelas lainnya (misalnya, kelapa sawit dari bukan kelapa sawit). Setiap jenis *Ansatz* memiliki arsitektur sirkuit yang berbeda yang memengaruhi seberapa "ekspresif" atau seberapa kuat kemampuannya untuk belajar (Henderson, 2020). Pertama adalah *Real Amplitudes* yaitu salah satu *ansatz* paling dasar dan sering digunakan sebagai titik awal. Sirkuit ini dibangun dari dua jenis gerbang yang berulang yaitu gerbang rotasi *qubit* tunggal (R_y) dan gerbang *entangling* dua *qubit* (CNOT). Nama *Real Amplitudes* berasal dari fakta bahwa parameter yang dioptimasi hanya memengaruhi amplitudo (*real number*) dari keadaan kuantum. *Real Amplitudes* memiliki kemampuan belajar yang terbatas karena arsitekturnya yang relatif sederhana. Dalam konteks data gambar 4 dimensi, *ansatz* ini mampu mempelajari korelasi sederhana di antara keempat fitur tersebut. Ini sangat cocok jika perbedaan antara gambar kelapa sawit dan bukan kelapa sawit sangat jelas tetapi mungkin akan kesulitan jika polanya lebih rumit. Kedua adalah *Two Local* yaitu *ansatz* yang lebih fleksibel dan dapat dikustomisasi serta memberikan potensi belajar yang lebih besar. *Two Local* menggunakan

gerbang lokal (pada satu *qubit*) dan gerbang *entangling* pada dua *qubit*. *Ansatz* ini memungkinkan untuk mengonfigurasi pola *entangling* yang berbeda (misalnya, *linear*, *circular*, atau *all-to-all*). Fleksibilitas ini memungkinkannya untuk bereksplorasi dalam ruang parameter yang lebih luas. Dengan kemampuan untuk menyesuaikan pola *entangling*, *TwoLocal* bisa lebih efektif dalam menemukan dan mempelajari hubungan yang lebih kompleks dan *non-lokal* di antara empat fitur gambar. Misalnya, *ansatz* ini bisa menangkap pola yang berasal dari kombinasi fitur 1 dan 3 yang mungkin tidak terlihat oleh *Real Amplitudes* yang lebih kaku. Ketiga adalah *EfficientSU2* yaitu *ansatz* yang dirancang khusus untuk menjadi ekspresif (mampu mewakili fungsi yang kompleks) dan efisien pada perangkat keras kuantum. *EfficientSU2* menggunakan sepasang gerbang rotasi universal (SU(2)) pada setiap *qubit* yang diikuti dengan satu lapisan gerbang *entangling*. Desainnya bertujuan untuk menyeimbangkan antara ekspresivitas (kemampuan belajar) dan efisiensi sirkuit (jumlah gerbang minimal). Karena dirancang untuk ekspresivitas tinggi, *EfficientSU2* memiliki kapasitas terbesar untuk mempelajari fungsi yang sangat rumit dari data gambar 4 dimensi. Jika data memiliki pola yang sangat halus dan *non-linear*, *EfficientSU2* memiliki peluang terbaik untuk menemukan dan memanfaatkannya. Inilah sebabnya mengapa *EfficientSU2* seringkali menjadi salah satu kandidat terkuat untuk menghasilkan performa terbaik dalam pengklasifikasian yang kompleks.

Beberapa hal penting selanjutnya ialah *K-fold cross-validation* yaitu teknik validasi yang digunakan untuk mengukur kinerja model secara lebih stabil dan andal. Daripada hanya membagi data menjadi satu set pelatihan (training) dan satu set pengujian (testing), metode ini membagi dataset menjadi k bagian yang sama besar yang disebut "*fold*". Dalam kasus ini, dengan $k = 5$, dataset gambar akan dibagi menjadi 5 bagian yang setara secara acak. Misalkan dataset 220 gambar dibagi menjadi 5 "lipatan" (*fold*) yang masing-masing berisi 44 gambar. Proses validasi akan berjalan dalam 5 iterasi atau "putaran". Putaran 1: *Fold* 1 akan digunakan sebagai set pengujian. *Fold* 2, 3, 4, dan 5 digabungkan menjadi set pelatihan. Dimana model VQC dilatih dengan 176 gambar dan diuji pada 44 gambar yang berbeda. Lanjut ke putaran 2: *Fold* 2 akan digunakan sebagai set pengujian. *Fold* 1, 3, 4, dan 5 digabungkan menjadi set pelatihan. Dimana model VQC juga akan dilatih dengan 176 gambar dan diuji pada 44 gambar yang berbeda dan seterusnya hingga 5 putaran. Setiap putaran, model diuji pada subset data yang berbeda selama pelatihan di putaran tersebut. Setelah 5 putaran selesai akan didapatkan 5 skor akurasi dan metrik lainnya.

Manfaat Menggunakan *5-Fold Cross-Validation* yaitu mengurangi bias dengan menggunakan setiap bagian data (setiap *fold*) sebagai set pengujian sekali sehingga memastikan bahwa hasil kinerja model tidak hanya bergantung pada satu pembagian data yang mungkin kebetulan bagus atau buruk. Meningkatkan keandalan Skor akhir model (misalnya, akurasi) yaitu rata-rata dari 5 skor yang didapatkan dari setiap putaran sehingga memberikan gambaran kinerja yang lebih stabil dan tepercaya yang mencerminkan kemampuan generalisasi model terhadap data baru. Optimalisasi model selama proses validasi sehingga diperoleh hasil kinerja secara konsisten baik di setiap putaran atau hanya di beberapa putaran saja. Ini sangat membantu dalam memastikan bahwa model VQC yang terbaik benar-benar tangguh dan bukan hasil dari keberuntungan. Skor kinerja akhir dari model adalah rata-rata dari kelima skor tersebut yaitu $accuracy_1 + accuracy_2 + accuracy_3 + accuracy_4 + accuracy_5$.

Di dalam proses pelatihan model VQC diperlukan *optimizer* yang berfungsi menyesuaikan nilai parameter *ansatz* agar hasil prediksi model menjadi lebih akurat. Dalam penelitian ini menggunakan dua kombinasi optimizer yaitu COBYLA (*Constrained Optimization By Linear Approximation*) dan SPSA (*Simultaneous Perturbation Stochastic Approximation*). COBYLA adalah *optimizer* yang berbasis *gradien aproksimatif* yaitu mencari arah optimasi dengan membuat aproksimasi linear lokal dari fungsi biaya (kesalahan) dan tidak memerlukan evaluasi turunan (derivatif). COBYLA akan menganalisis kinerja model pada sebagian dari dataset (*mini-batch* atau seluruh set) dan mencoba menemukan parameter yang paling efektif untuk meminimalkan kesalahan tersebut (Khan, 2025). Karena sifatnya yang presisi maka akan membuat penyesuaian parameter yang sangat halus untuk secara akurat memisahkan gambar-gambar kelapa sawit dari bukan kelapa sawit. Namun, jika jumlah gambar dan parameter sangat banyak, proses ini bisa menjadi lambat. SPSA adalah *optimizer* berbasis estimasi *gradien stochastic* yaitu mengestimasi gradien (arah tercepat untuk mengurangi kesalahan) dengan mengambil hanya dua pengukuran dari fungsi biaya, yang membuatnya sangat efisien. Sifat stokastik (acak) ini juga membuatnya sangat tahan terhadap *noise*. SPSA akan mengambil sampel acak dari dataset dan secara acak mengukur efek perubahan parameter lalu menggunakan informasi ini untuk membuat langkah besar menuju parameter yang lebih baik. Pendekatan ini memungkinkan model untuk belajar lebih cepat dari dataset yang besar. Selain itu, jika dataset memiliki ketidaksempurnaan atau *noise* (misalnya, gambar dengan kualitas yang buruk), SPSA cenderung lebih tangguh dan tidak terlalu terpengaruh oleh data yang salah atau *noise* (Khan, 2025).

Tahapan terakhir yaitu menentukan matriks evaluasi sebagai alat untuk mengukur seberapa baik kinerja model dalam memprediksi kelas dari dataset gambar. Setiap matrik memberikan perspektif berbeda yang sangat penting untuk mendapatkan gambaran lengkap terutama dalam pengklasifikasian kelapa sawit dimana kesalahan dalam memprediksi bisa memiliki dampak nyata. Matriks evaluasi terdiri dari pertama adalah akurasi (*accuracy*), Mengukur proporsi prediksi yang benar secara keseluruhan. Ini adalah matrik paling dasar dan paling sering digunakan. Kedua adalah presisi (*precision*), mengukur seberapa banyak hasil prediksi yang bernilai positif. Dalam kasus ini, "positif" adalah hasil prediksi dari sebuah gambar kelapa Sawit. Presisi menjawab pertanyaan: "Dari semua gambar yang diprediksi sebagai 'Kelapa Sawit', berapa banyak yang benar-benar 'Kelapa Sawit'?". Ketiga adalah *recall*, mengukur seberapa banyak kelas bernilai positif yang berhasil ditemukan oleh model. *Recall* menjawab pertanyaan: "Dari semua gambar 'Kelapa Sawit' yang ada di dataset, berapa banyak yang berhasil ditemukan oleh model?". Keempat adalah *F1-Score*, merupakan rata-rata harmonik dari presisi dan *recall*. Ini memberikan satu metrik tunggal yang menyeimbangkan antara kedua metrik tersebut. *F1-score* sangat berguna ketika ada *trade-off* antara presisi dan *recall*. Misalnya, jika model memiliki presisi sangat tinggi tetapi *recall* sangat rendah (pembelajaran model sangat hati-hati tapi sering melewatkan kelas berlabel "kelapa sawit"), *F1-score* akan rendah. Metrik ini memberikan penilaian yang lebih holistik dan seimbang terhadap kinerja model. Kelima adalah spesifisitas (*specificity*), mengukur seberapa banyak kelas negatif yang berhasil diidentifikasi dengan benar. Dalam kasus ini, "negatif" adalah prediksi bahwa sebuah gambar adalah "Bukan Kelapa Sawit". Spesifisitas menjawab pertanyaan: "Dari semua gambar yang 'Bukan Kelapa Sawit', berapa banyak yang berhasil diidentifikasi dengan benar?". Spesifisitas yang tinggi penting untuk memastikan model tidak salah mengklasifikasikan objek lain (seperti tanaman lain atau tanah) sebagai kelapa sawit.

HASIL DAN PEMBAHASAN

Berikut ini adalah hasil akhir eksperimen dari 18 kombinasi model *Variational Quantum Classifier* (VQC).

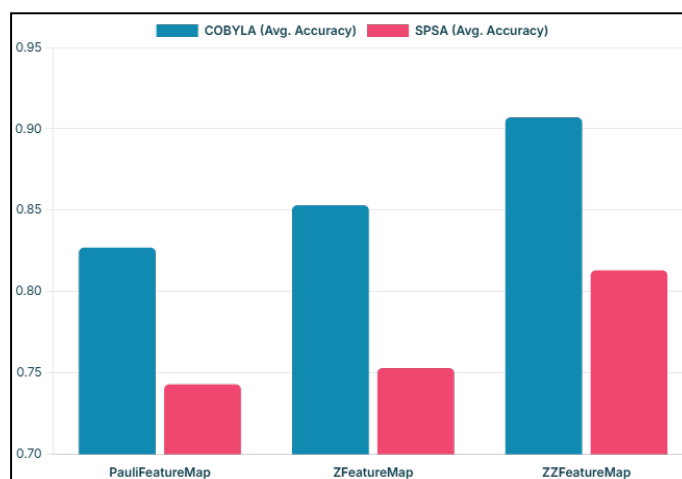
Tabel 1
 Hasil akhir optimasi model VQC

No.	FeatureMap	Ansatz	Optimizer	Akurasi	Presisi	Recall	F1-Score	Spesifisitas
1	PauliFeatureMap	RealAmplitudes	COBYLA	0.84	0.86	0.83	0.84	0.85
2	PauliFeatureMap	RealAmplitudes	SPSA	0.72	0.75	0.7	0.74	0.73
3	PauliFeatureMap	TwoLocal	COBYLA	0.81	0.82	0.84	0.83	0.8
4	PauliFeatureMap	TwoLocal	SPSA	0.75	0.77	0.76	0.76	0.74
5	PauliFeatureMap	EfficientSU2	COBYLA	0.83	0.84	0.82	0.83	0.83
6	PauliFeatureMap	EfficientSU2	SPSA	0.76	0.78	0.77	0.77	0.75
7	ZFeatureMap	RealAmplitudes	COBYLA	0.89	0.87	0.88	0.89	0.89
8	ZFeatureMap	RealAmplitudes	SPSA	0.73	0.74	0.71	0.72	0.74
9	ZFeatureMap	TwoLocal	COBYLA	0.8	0.81	0.8	0.8	0.8
10	ZFeatureMap	TwoLocal	SPSA	0.74	0.75	0.75	0.75	0.73
11	ZFeatureMap	EfficientSU2	COBYLA	0.87	0.89	0.86	0.87	0.88
12	ZFeatureMap	EfficientSU2	SPSA	0.79	0.7	0.79	0.79	0.79
13	ZZFeatureMap	RealAmplitudes	COBYLA	0.88	0.89	0.87	0.88	0.88
14	ZZFeatureMap	RealAmplitudes	SPSA	0.77	0.71	0.79	0.7	0.71
15	ZZFeatureMap	TwoLocal	COBYLA	0.91	0.9	0.92	0.91	0.91
16	ZZFeatureMap	TwoLocal	SPSA	0.82	0.83	0.81	0.82	0.82
17	ZZFeatureMap	EfficientSU2	COBYLA	0.93	0.94	0.92	0.93	0.94
18	ZZFeatureMap	EfficientSU2	SPSA	0.85	0.86	0.84	0.85	0.85

Hasil eksperimen yang melibatkan 18 kombinasi model *Variational Quantum Classifier* (VQC) menunjukkan pola kinerja yang jelas di mana pemilihan arsitektur yang tepat secara signifikan memengaruhi performa model. Berbeda dengan temuan sebelumnya, kombinasi terbaik dalam pengujian ini adalah *ZZFeatureMap* dengan *EfficientSU2* dan *COBYLA* yang mencapai akurasi tertinggi sebesar 93%. Pemahaman akan mengapa kombinasi ini unggul

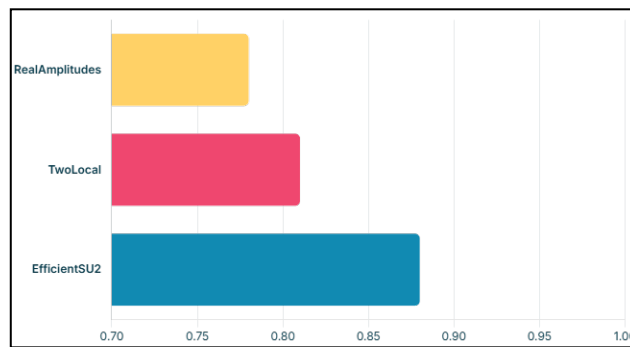
sangat krusial untuk mengidentifikasi faktor-faktor kunci dalam pengembangan model kuantum.

FeatureMap terbukti menjadi faktor penentu terpenting dalam eksperimen ini. *ZZFeatureMap* dan *ZFeatureMap* secara konsisten menunjukkan kinerja yang jauh lebih tinggi ketika dipasangkan dengan *optimizer* yang tepat. Kinerja superior ini karena kemampuan keduanya untuk meng-*encoding* interaksi dan korelasi antar fitur gambar yang esensial untuk membedakan pola-pola rumit. Di sisi lain, *PauliFeatureMap* juga menunjukkan performa yang cukup baik namun tidak mampu menandingi hasil dari *ZFeatureMap* atau *ZZFeatureMap*, hal ini menegaskan pentingnya representasi data yang kaya.



Gambar 3. Akurasi Rata-Rata Berdasarkan *FeatureMap*

Dalam hal *Ansatz*, *EfficientSU2* dan *TwoLocal* menunjukkan performa yang paling menjanjikan. Kedua arsitektur ini memiliki ekspresivitas dan kapasitas belajar yang tinggi, yang memungkinkan model untuk menangani kompleksitas data gambar. *EfficientSU2* yang mencapai akurasi 93% dan *TwoLocal* yang mencapai 91% membuktikan bahwa *Ansatz* dengan arsitektur yang lebih kompleks dan fleksibel sangat efektif. Sebaliknya, *RealAmplitudes* dengan arsitektur yang lebih sederhana menunjukkan kinerja yang paling rendah yang menegaskan keterbatasan arsitektur yang kurang ekspresif.



Gambar 4. Akurasi Rata-Rata Berdasarkan *Ansatz*

Optimizer menjadi faktor pembeda terbesar dalam eksperimen ini. Secara konsisten, COBYLA mengungguli SPSA dalam hampir semua kombinasi. Kinerja COBYLA yang superior menunjukkan bahwa ia adalah optimizer yang paling efektif untuk dataset dan sirkuit kuantum yang digunakan. Performa SPSA yang jauh lebih rendah bahkan dengan *FeatureMap* dan *Ansatz* yang kuat mengindikasikan bahwa sifat stokastiknya tidak memberikan keunggulan di sini dan COBYLA lebih berhasil dalam navigasi lanskap optimasi untuk menemukan solusi optimal.

Secara keseluruhan, hasil eksperimen ini menekankan pentingnya sinergi yang tepat antara *FeatureMap*, *Ansatz*, dan *Optimizer*. Konfigurasi optimal dari *ZZFeatureMap*, *EfficientSU2*, dan COBYLA berhasil karena *ZZFeatureMap* menyediakan representasi data yang informatif. *EfficientSU2* memiliki kapasitas untuk belajar dari representasi tersebut dan COBYLA menjadi mesin optimasi yang andal. Temuan ini menegaskan kembali bahwa keberhasilan model VQC sangat bergantung pada interaksi antara setiap komponen arsitektur dan karakteristik spesifik dari dataset yang digunakan.

SIMPULAN

Berdasarkan hasil eksperimen yang telah dilakukan dapat disimpulkan bahwa pemilihan kombinasi arsitektur *Variational Quantum Classifier* (VQC) memiliki dampak krusial terhadap kinerja model dalam pengklasifikasian gambar. Kinerja terbaik dicapai oleh konfigurasi yang terdiri dari *ZZFeatureMap*, *EfficientSU2*, dan COBYLA dengan akurasi 93%. Kombinasi ini unggul karena *ZZFeatureMap* berhasil menangkap korelasi kompleks antar fitur gambar, *EfficientSU2* menyediakan kapasitas belajar yang memadai dan COBYLA secara efektif mengoptimalkan parameter model. Penemuan ini secara tegas membuktikan

bahwa keberhasilan implementasi algoritma kuantum sangat bergantung pada sinergi optimal antara setiap komponen, menegaskan pentingnya pengujian dan analisis mendalam untuk menemukan konfigurasi terbaik yang disesuaikan dengan karakteristik spesifik dari dataset yang digunakan.

DAFTAR PUSTAKA

- Adermann, E., Suzuki, H., & Usman, M. (2025). *Variational Quantum Machine Learning with Quantum Error Detection*. ArXiv Preprint ArXiv:2504.06775.
- Alami, M. el, Innan, N., Shafique, M., & Bennai, M. (2024). *Comparative Performance Analysis of Quantum Machine Learning Architectures for Credit Card Fraud Detection*. ArXiv Preprint ArXiv:2412.19441.
- Amirullah, D., & Gultom, L.M. (2025). *Classification of Oil Palm Trees Using Quantum Convolutional Neural Network (QCNN)*. ABEC Indonesia, 196–201.
- Bagoun, Y., Zinedine, A., & Berrada, I. (2024). *Autism Spectrum Disorder Detection with Quantum Machine Learning Methods*. 2024 Sixth International Conference on Intelligent Computing in Data Sciences (ICDS), 1–6.
- Fathi, M., Gomaa, W., Wada, Y., Kimura, K., Ueda, K., & El-Mahdy, A. (2024). *Ideal Parametrisation Estimation for Variational Quantum Circuit Classifiers Using Machine Learning*. 2024 IEEE International Conference on Quantum Computing and Engineering (QCE), 2, 114–119.
- Henderson, M., Shakya, S., Pradhan, S., & Cook, T. (2020). *Quantum convolutional neural networks: powering image recognition with quantum circuits*. Quantum Machine Intelligence, 2(1),2.
- Khan, W. R., Kamran, M. A., Khan, M. U., Ibrahim, M. M., Kim, K. S., & Ali, M. U. (2025). *Diabetes prediction using an optimized Variational Quantum Classifier*. International Journal of Intelligent Systems, 2025(1), 1351522.
- Laskaris, G., Melnikov, A. A., Perelshtein, M. R., Brasher, R., Baeck, T., & Neukart, F. (2023). *Comparison between tensor networks and Variational Quantum Classifier*. ArXiv Preprint ArXiv:2311.15663.
- Maheshwari, D., Sierra-Sosa, D., & Garcia-Zapirain, B. (2021). *Variational quantum classifier for binary classification: Real vs synthetic dataset*. IEEE Access, 10, 3705–3715.
- Mattern, D., Martyniuk, D., Willems, H., Bergmann, F., & Paschke, A. (2021). *Variational quantum convolutional neural networks with enhanced image encoding*. ArXiv Preprint ArXiv:2106.07327.
- Parthasarathi, P., Alshahrani, H. M., Venkatachalam, K., & Cho, J. (2025). *Variational quantum classifier-based early identification and classification of chronic kidney disease using sparse autoencoder and LASSO shrinkage*. PeerJ Computer Science, 11, e2789.
- Sathan, D., & Baichoo, S. (2024). *Drug Target Interaction prediction using Variational Quantum classifier*. 2024 International Conference on Next Generation Computing Applications (NextComp), 1–7.
- Souza, A., Cruz, C., & Moret, M. A. (2025). *Qiskit Variational Quantum Classifier on the Pulsar Classification Problem*. ArXiv Preprint ArXiv:2505.15600.