

KOMPARASI WAKTU PEMROSESAN CPU ANTARA QUANTUM HADAMARD EDGE DETECTION (QHED), CANNY DAN SOBEL DALAM DETEKSI TEPI GAMBAR PERKEBUNAN KELAPA SAWIT**Lipantri Mashur Gultom¹⁾, Desi Amirullah²⁾**

^{1,2} Jurusan Teknik Informatika, Politeknik Negeri Bengkalis, Jl. Bathin Alam, Sungai Alam, Bengkalis, Riau, 28714
E-mail: lipantri@polbeng.ac.id

Abstract

This research was conducted to compare the CPU processing time between Quantum Hadamard Edge Detection (QHED), Canny, and Sobel in Palm Oil Plantation Image Edge Detection with several pixel resolution variations. In QHED, several qubits are used, namely 2, 4, 6, 8, 10, and 12 qubits, while the number of qubits greater than 12 cannot be tested due to the limited RAM memory of the device in this study. The final result of this research proves that QHED can detect the edge of an image with the fastest processing time at the number of qubits 6, while the best image edge detection quality lies at the number of qubits 2. In addition, the comparison of QHED, Canny, and Sobel results in CPU processing time where Canny and Sobel are still faster than QHED, but the quality of image edge detection is still better in QHED.

Keywords: *Quantum, Hadamard, Edge Detection*

Abstrak

Penelitian ini dilakukan untuk membandingkan waktu pemrosesan CPU antara Quantum Hadamard Edge Detection (QHED), Canny Dan Sobel Dalam Deteksi Tepi Gambar Perkebunan Kelapa Sawit dengan beberapa variasi resolusi pixel. Pada QHED digunakan beberapa jumlah qubit yaitu 2, 4, 6, 8, 10 dan 12 qubit, sedangkan jumlah qubit lebih dari 12 tidak dapat diuji karena keterbatasan memori RAM dari perangkat yang ada dalam penelitian ini. Hasil akhir dari penelitian ini membuktikan bahwa QHED dapat mendeteksi tepi suatu gambar dengan waktu pemrosesan yang paling cepat pada jumlah qubit 6 sedangkan kualitas deteksi tepi gambar yang terbaik terletak pada jumlah qubit 2. Selain itu perbandingan QHED, Canny dan Sobel menghasilkan waktu pemrosesan CPU dimana Canny dan Sobel masih lebih cepat dibandingkan QHED namun kualitas deteksi tepi gambar masih lebih bagus pada QHED.

Kata Kunci: *Quantum, Hadamard, Edge Detection*

PENDAHULUAN

Indonesia merupakan salah satu negara pengekspor hasil perkebunan sawit (sumber : <https://www.liputan6.com>, <https://www.wartasawit.com>, <https://dunia.tempo.co>) terbanyak di dunia. Hal ini tidak dapat dimungkiri karena pada kenyataannya beberapa wilayah di Indonesia merupakan lahan perkebunan sawit yang sangat luas seperti pulau Sumatera dan Kalimantan. Beberapa hasil kajian deteksi tepi gambar khususnya pada topik perkebunan sawit (Eja, 2021)(Irsanti, 2019)(Prasvita, 2021)(Rosalina, 2019)(Srinarta, 2022) masih berfokus pada metode klasik dengan memaparkan beberapa kelebihan dan kekurangannya masing-masing. Perkembangan teknologi komputasi sekarang ini yang sudah mulai menerapkan komputasi kuantum, peneliti menfokuskan pada penerapan deteksi tepi gambar dengan pendekatan komputasi kuantum.

Quantum Hadamard Edge Detection (QHED) merupakan metode lain dalam pengolahan citra yang menggunakan pendekatan komputasi kuantum dalam representasi dan penyimpanan data citra (Yan, 2020), (Yan, 2017), (Pramanik, 2021). Dalam komputasi kuantum semua nilai keadaan disimpan kedalam bentuk *qubit* dimana biasanya hasil akhir pemrosesan bergantung pada rangkaian gerbang logika kuantum (Gan, 2022) (Anand, 2022). QHED merupakan salah satu algoritma pendeteksian tepi dengan pendekatan komputasi kuantum menggunakan transformasi *hadamard*. Pada QHED memiliki skema pengkodean gambar yang efisien dalam menghemat ruang memori dengan menggunakan pengkodean amplitudo terhadap penurunan eksponensial dalam jumlah *qubit* yang digunakan yaitu $n = \log_2 N$ (Geng, 2022). Namun kompleksitas waktu dalam persiapan awal pada pengkodean gambar sangatlah tinggi. Oleh karena itu penelitian ini difokuskan pada penganalisisan bagaimana pengaruh jumlah *qubit* yang digunakan pada QHED terhadap waktu dan ruang memori yang digunakan (Ruan, 2021)(Su, 2020).

QHED menggunakan gerbang *Hadamard* (H) sebagai operasi transformasi *qubit* (Yuan, 2019).

$$|0\rangle = \frac{(|0\rangle + |1\rangle)}{\sqrt{2}} \text{ dan } |1\rangle = \frac{(|0\rangle - |1\rangle)}{\sqrt{2}} \quad (1)$$

kemudian *N-pixel* gambar yang diproses akan diberi nomor dengan menggunakan *binari bit-string* $|b_{n-1}b_{n-2}b_{n-3} \dots b_1b_0\rangle$ dimana $b_i \in \{0,1\}$. Untuk dua *pixel* yang berdekatan, maka dapat dituliskan dalam bentuk pasangan *bit-string* $|b_{n-1}b_{n-2} \dots b_10\rangle$

dan $|b_{n-1}b_{n-2} \dots b_1\rangle$ dimana hanya *least significant bit* (LSB) yang berbeda dari keduanya. Masing – masing nilai intensitas *pixel* yang sesuai (dinormalisasi) dapat ditulis sebagai $c_{b_{n-1}b_{n-2} \dots b_1 0}$ dan $c_{b_{n-1}b_{n-2} \dots b_1 1}$. Kemudian penulisan nilai *pixel* tersebut dapat disederhanakan kedalam representasi desimal yaitu c_i dan c_{i+1} . Selanjutnya penerapan gerbang H ke LSB pada ruang register kuantum menjadi hasil unitari sebagai berikut.

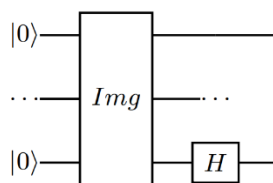
$$I_{2^{n-1}} \otimes H_0 = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 & 0 & 0 & \dots & 0 & 0 \\ 1 & -1 & 0 & 0 & \dots & 0 & 0 \\ 0 & 0 & 1 & 1 & \dots & 0 & 0 \\ 0 & 0 & 1 & -1 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \dots & 1 & 1 \\ 0 & 0 & 0 & 0 & \dots & 1 & -1 \end{bmatrix} \quad (2)$$

dimana $I_{2^{n-1}}$ adalah *matrix* $2^{n-1} \times 2^{n-1}$. Hasil unitari yang berisikan nilai *pixel* gambar kedalam register kuantum dikodekan menggunakan *Quantum Probability Image Encoding* (QPIE)

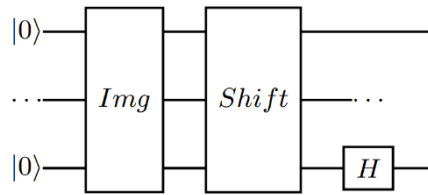
$$|Img\rangle = \sum_{i=0}^{N-1} c_i |i\rangle \quad (3)$$

$$(I_{2^{n-1}} \otimes H_0) \cdot \begin{bmatrix} c_0 \\ c_1 \\ c_2 \\ c_3 \\ \vdots \\ c_{N-2} \\ c_{N-1} \end{bmatrix} \rightarrow \frac{1}{\sqrt{2}} \begin{bmatrix} c_0 + c_1 \\ c_0 - c_1 \\ c_2 + c_3 \\ c_2 - c_3 \\ \vdots \\ c_{N-2} + c_{N-1} \\ c_{N-2} - c_{N-1} \end{bmatrix} \quad (4)$$

Berikut gambar *circuit* untuk mengekstrak semua informasi tepi dari gambar (Cavalieri, 2020).

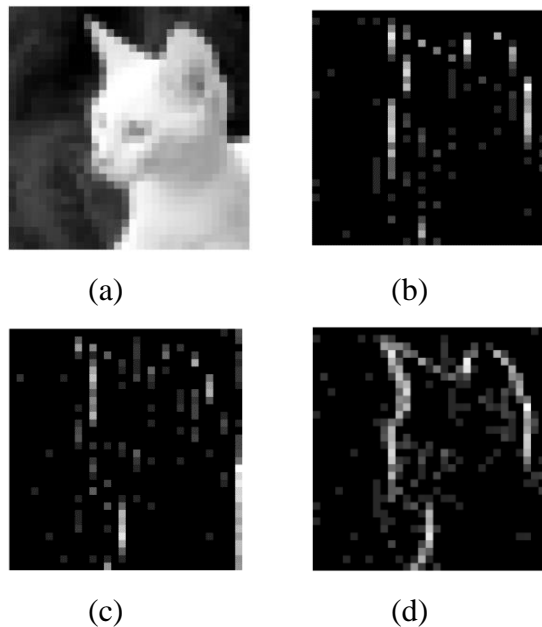


Gambar 1. Gerbang logika untuk mendapatkan nilai $(c_0 - c_1)$, $(c_2 - c_3)$, ...



Gambar 2. Gerbang logika untuk mendapatkan nilai $(c_1 - c_2)$, $(c_3 - c_4)$, ...

dari hasil *matrix* persamaan (4) dapat dihasilkan deteksi tepi secara horizontal antara pasangan *pixel* bilangan genap : 0 & 1, 2 & 3 , dst serta deteksi tepi secara horizontal antara pasangan *pixel* bilangan ganjil : 1 & 2, 2 & 3, dst. Sehingga untuk memperoleh nilai permutasi amplitudo pada register kuantum dengan mengkonversi vektor amplitudo $(c_0, c_1, c_2, \dots, c_{N-1})^T$ menjadi $(c_1, c_2, c_3, \dots, c_{N-1})^T$. Setelah itu ditransformasi dengan gerbang H dan dilakukan pengukuran register kuantum pada LSB sampai menjadi **11**). Secara umum dapat dilihat contoh pemrosesan gambar berikut ini (Cavalieri, 2020).

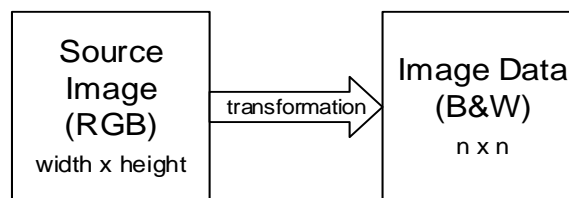


Gambar 3. Contoh (a) sumber gambar, (b) hasil gambar pemrosesan dari fig. 1, (c) hasil gambar pemrosesan dari fig. 2 dan (d) penggabungan fig. 3, fig. 4 dan fig. 5

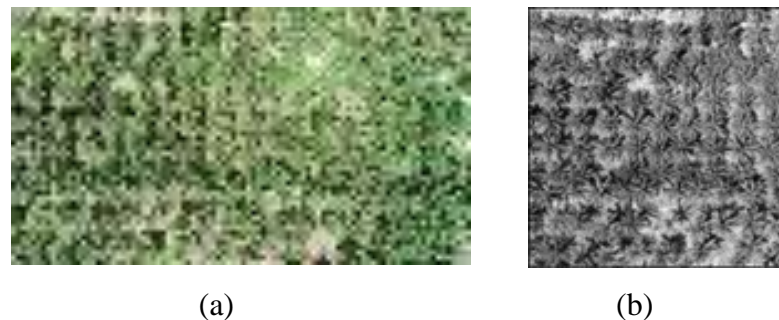
METODE PENELITIAN

Bab ini fokus pada pengimplementasian seluruh tahapan yang sudah diuraikan pada bab sebelumnya. Sedangkan *dataset* gambar yang diuji menggunakan gambar perkebunan sawit yang diperoleh dari *drone* merek DJI phantom 4. Kode program dibangun dengan menggunakan *Qiskit* dan disimulasikan dengan *Qiskit backend statevector simulator*. Perangkat keras yang digunakan yaitu Dell inspiron 3881 dengan RAM memory 16 GB (DDR4 SDRAM), Processor Intel(R) Core (TM) i7-10700F CPU @ 2.90 GHz (8 Core) dan Sistem Operasi Windows 10 Home Single Language 64 Bit.

Pertama sekali dilakukan tahapan pre-pemrosesan data gambar yang akan diproses kedalam QHED (Pramanik, 2021).

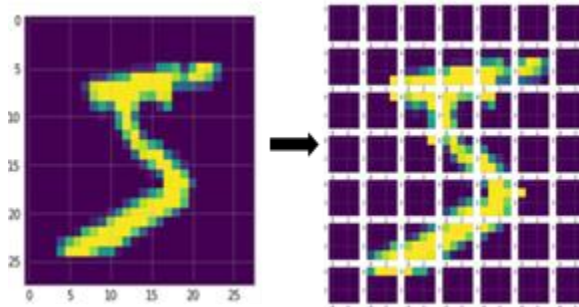


Gambar 4. Pre-pemrosesan data gambar



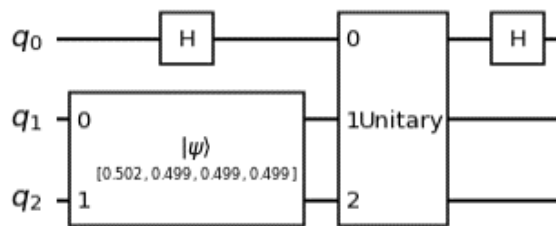
Gambar 5. Contoh pre-pemrosesan data gambar : (a) gambar RGB dengan 481 x 321 *pixel* dan (b) gambar B&W image dengan 321 x 321 *pixel*

kemudian dilakukan pemilihan pasangan jumlah *qubit* (*N-qubit*) dan ukuran *chunk* 2^{cp} untuk membagi $n \times n$ *pixel* gambar menjadi $2^{cp} \times 2^{cp}$ (Anand, 2022).

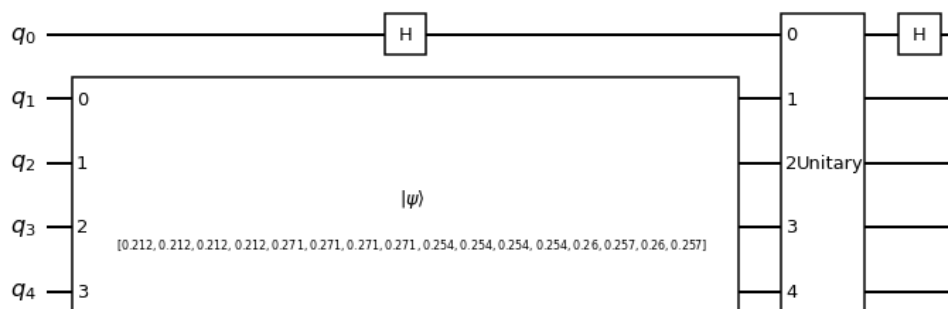


Gambar 6. Pembagian $n \times n$ gambar kedalam $2^{cp} \times 2^{cp}$

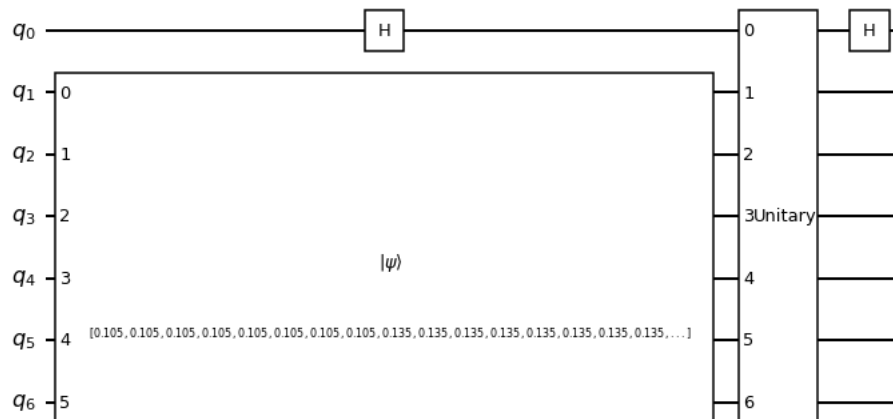
jumlah *qubit* disini digunakan untuk merancang quantum *circuit* dari fig.7. Jadi misalkan dipilih pasangan (2, 1) *N-qubit* = 2 dan $cp = 1$ maka ukuran *chunk* yang sesuai yaitu $2^{cp} = 2^1 = 2$. dan seterusnya dimana *N-qubit* bertambah 2 dan cp bertambah 1 menjadi (4,2), (6,3), (8, 4) dan seterusnya.



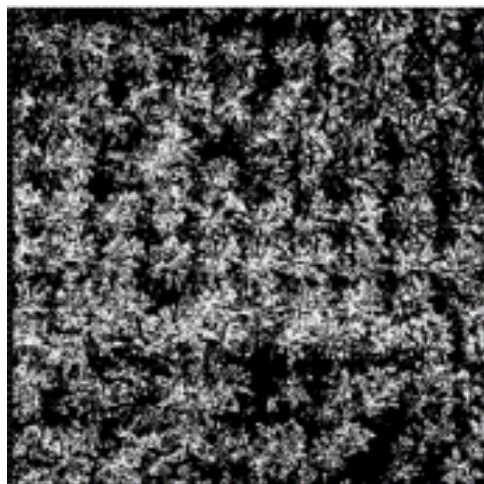
Gambar 7. Contoh rangkaian QHED dengan 2 *qubit*



Gambar 8. Contoh rangkaian QHED dengan 4 *qubit*



Gambar 9. Contoh rangkaian QHED dengan 6 *qubit* dengan contoh hasil akhir dari proses QHED dapat dilihat pada gambar berikut



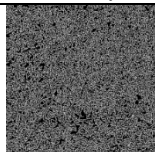
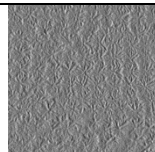
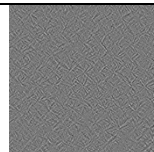
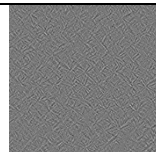
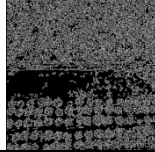
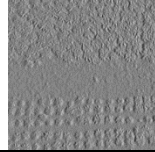
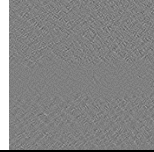
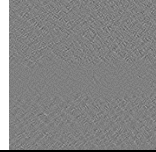
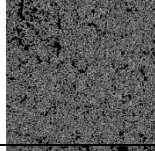
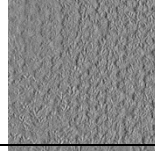
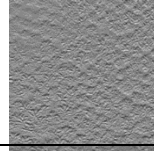
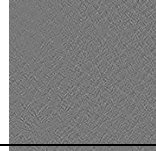
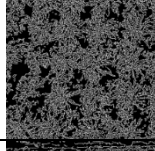
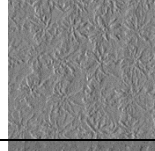
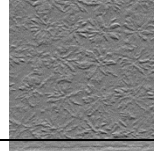
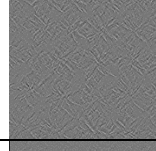
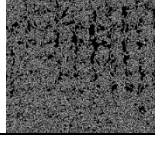
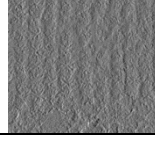


Gambar 10. Contoh hasil dari QHED

HASIL DAN PEMBAHASAN

Rangkuman hasil pengujian dengan beberapa dataset gambar yang digunakan dapat dilihat pada tabel 1 pada lampiran. Secara keseluruhan dari pengujian beberapa gambar tersebut dapat dilihat hasil segmentasi terbanyak dan terbaik terletak pada proses deteksi tepi gambar jika menggunakan 2 *qubit*. sementara itu hasil yang terburuk jika menggunakan 12 *qubit*. Pada penelitian ini juga dilakukan pengujian dengan 14 *qubit* namun dengan spesifikasi memori (RAM 16 GB) tidak mampu mengalokasikan semua

data yang dibutuhkan untuk hasil pemrosesan. Disini juga ditampilkan hasil deteksi tepi dengan metode yang tidak menggunakan pendekatan komputasi kuantum yaitu metode canny (Canny, 1986) dan sobel (Irwin, 1968) dapat dilihat tabel 2 berikut ini.

Tabel 2.
Hasil deteksi tepi dengan metode canny dan sobel

No.	Canny	Sobel X	Sobel Y	Sobel XY
1				
2				
3				
4				
5				

dapat dibandingkan bahwa hasil dengan metode canny lebih mendekati dengan hasil QHED daripada dengan metode sobel. Berikutnya hasil pengujian dari pengaruh jumlah *qubit* terhadap waktu pemrosesan CPU dapat dilihat pada tabel 3 berikut ini.

Tabel 3.
Pengaruh jumlah *qubit* terhadap waktu pemrosesan CPU (dalam detik)

No.	QHED dengan jumlah <i>qubit</i>						canny	sobel
	2	4	6	8	10	12		
1	1193.88	156.85	49.87	71.56	508.38	7093.00	3.43	10.54
2	1163.39	158.28	50.04	71.93	504.19	7085.41	3.41	10.49
3	1163.97	163.21	49.86	74.27	505.60	7060.95	3.53	10.52
4	1199.65	159.29	53.03	73.02	508.01	7134.62	3.47	10.56
5	1213.50	150.59	49.46	71.64	505.57	7065.30	3.43	10.49

secara keseluruhan waktu pemrosesan QHED yang paling cepat dengan penggunaan 6 *qubit* dan yang paling lama dengan penggunaan 12 *qubit*. Jika diperhatikan secara keseluruhan ternyata semakin banyak jumlah *qubit* yang digunakan maka semakin lama juga waktu pemrosesannya namun pada tabel 3 ditemukan ada sesuatu yang berbeda yaitu waktu pemrosesan dengan menggunakan 2 *qubit* relatif lebih lama dibandingkan dengan jumlah *qubit* sebanyak 4, 6, 8 dan 10. Hal ini bisa terjadi demikian karena pada tahapan QHED terdapat proses pembagian satu gambar menjadi beberapa kelompok gambar yang lebih kecil dengan pemotongan gambar menjadi $2^{cp} \times 2^{cp}$ seperti uraian pada gambar 7. selain itu metode canny dan sobel masih lebih cepat dalam waktu pemrosesan dibandingkan dengan QHED.

SIMPULAN

Hasil penelitian ini membuktikan bahwa secara teoritis QHED dapat melakukan proses deteksi tepi pada gambar namun dari pengujian yang dilakukan ternyata penggunaan jumlah *qubit* sangat berpengaruh pada waktu pemrosesan CPU dimana waktu pemrosesan tercepat terletak pada penggunaan jumlah *qubit* sebanyak 6 *qubit* bukan terletak pada jumlah *qubit* yang paling sedikit yaitu 2 *qubit*. Sedangkan hasil deteksi tepi yang terbaik pada penggunaan jumlah 2 *qubit*.

DAFTAR PUSTAKA

- Anand, A., Lyu, M., Baweja, P. S., & Patil, V. (2022). Quantum Image Processing. *arXiv preprint arXiv:2203.01831*.
- Cavaliere, G., & Maio, D. (2020). A quantum edge detection algorithm. *arXiv preprint arXiv:2012.11036*.
- Canny, J. (1986). A computational approach to edge detection. *IEEE Transactions on pattern analysis and machine intelligence*, (6), 679-698.
- Eja, I. R., & Basyid, M. (2021). Kajian Perhitungan Pohon Kelapa Sawit Menggunakan Foto Udara Dan Citra Satelit Pleiades Dengan Metode Obia (Object Based Image Analysis) (Studi Kasus Pulau Sebatik). *FTSP*, 518-524.
- F. G. Irwin and others. (1968). "An isotropic 3x3 image gradient operator," Present. Stanford AI Proj., vol. 2014, no. 02.

- Gan, W. S. (2022). Quantum image processing. In *Quantum Acoustical Imaging* (pp. 83-86). Springer, Singapore.
- Geng, A., Moghiseh, A., Redenbach, C., & Schladitz, K. (2022). A hybrid quantum image edge detector for the NISQ era. *arXiv preprint arXiv:2203.12072*.
- Irsanti, D., Sasmito, B., & Bashit, N. (2019). Kajian Pengaruh Penajaman Citra Untuk Penghitungan Jumlah Pohon Kelapa Sawit Secara Otomatis Menggunakan Foto Udara (Studi Kasus: KHG Bentayan Sumatra Selatan). *Jurnal Geodesi Undip*, 8(1), 428-434.
- Pramanik, S., Chandra, M. G., Sridhar, C. V., Kulkarni, A., Sahoo, P., DV, V. C., ... & Nambiar, M. (2021). A quantum-classical hybrid method for image classification and segmentation. *arXiv preprint arXiv:2109.14431*.
- Prasvita, D. S., Santoni, M. M., Wirawan, R., & Trihastuti, N. (2021). KLASIFIKASI POHON KELAPA SAWIT PADA DATA FUSI CITRA LIDAR DAN FOTO UDARA MENGGUNAKAN CONVOLUTIONAL NEURAL NETWORK. *JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)*, 6(2), 406-415.
- Elin, R., & Agustin, S. (2019). Klasifikasi Umur Lahan Perkebunan Kelapa Sawit Pada Citra Foto Udara Berdasarkan Tekstur Menggunakan Metode Naïve Bayes. *Indexia: Informatics and Computational Intelligent Journal*, 1(1), 6-13.
- Srinarta, K. (2022). Analisis Perhitungan Jumlah Pohon Kelapa Sawit Berdasarkan Algoritma Canopy Height Model (Chm) Dan Local Maxima (Lm). *Jurnal Geodesi UNDIP*, 11(1), 51-60.
- Su, J., Guo, X., Liu, C., & Li, L. (2020). A new trend of quantum image representations. *IEEE Access*, 8, 214520-214537.
- Yan, F., Venegas-Andraca, S.E. (2020). Quantum Image Operations. In: *Quantum Image Processing*. Springer, Singapore. https://doi.org/10.1007/978-981-32-9331-1_3
- Yan, F., Ilyasu, A. M., & Le, P. Q. (2017). Quantum image processing: a review of advances in its security technologies. *International Journal of Quantum Information*, 15(03), 1730001. *Int. J. Quantum Inf.*, vol. 15, no. 03, p. 1730001, 2017.
- Yuan, S., Venegas-Andraca, S. E., Wang, Y., Luo, Y., & Mao, X. (2019). Quantum image edge detection algorithm. *International Journal of Theoretical Physics*, 58(9), 2823-2833.
- <https://www.liputan6.com/bisnis/read/4575604/indonesia-jadi-eksportir-minyak-sawit-terbesar-di-dunia> (diakses tanggal 10 Mei 2022)
- <https://www.wartasawit.com/read/991/indonesia-kembali-jadi-eksportir-terbesar-minyak-sawit-dunia.html> (diakses tanggal 10 Mei 2022)



11th Applied Business and Engineering Conference

<https://bisnis.tempo.co/read/1469857/ri-ekspor-373-juta-ton-minyak-sawit-pada-2020-terbesar-di-dunia> (diakses tanggal 10 Mei 2022)

<https://www.ibm.com/quantum-computing/>

<https://qiskit.org/>