

Write a program to enhance the “flower1.jpg” and “fruit.jpg” images in frequency domain using the following filters:

1. Notch filters

a. Low-pass filter

$$H(u, v) = \begin{cases} 1, & \text{if } (u, v) \text{ is closer to the center than } r \\ 0, & \text{if } (u, v) \text{ is further from the center than } r \end{cases}$$

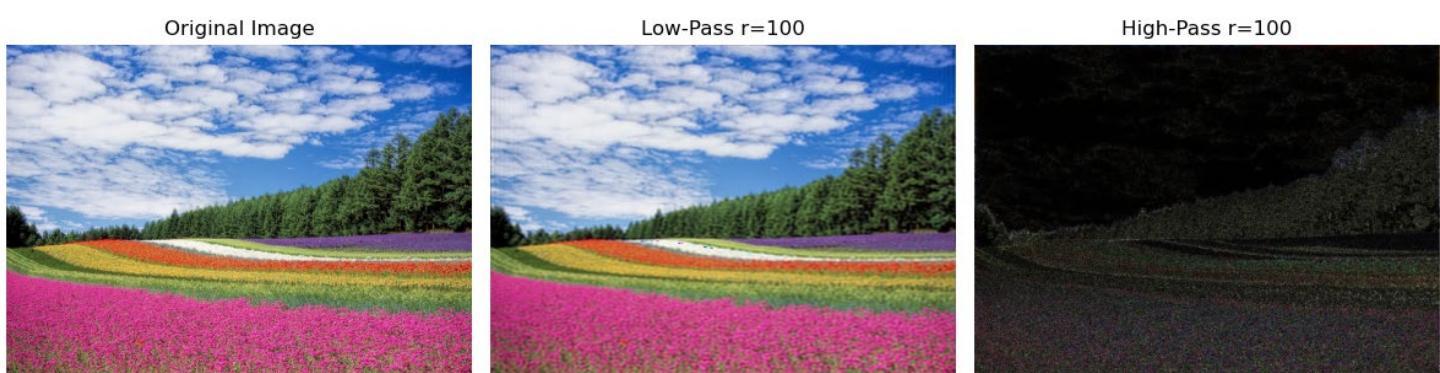
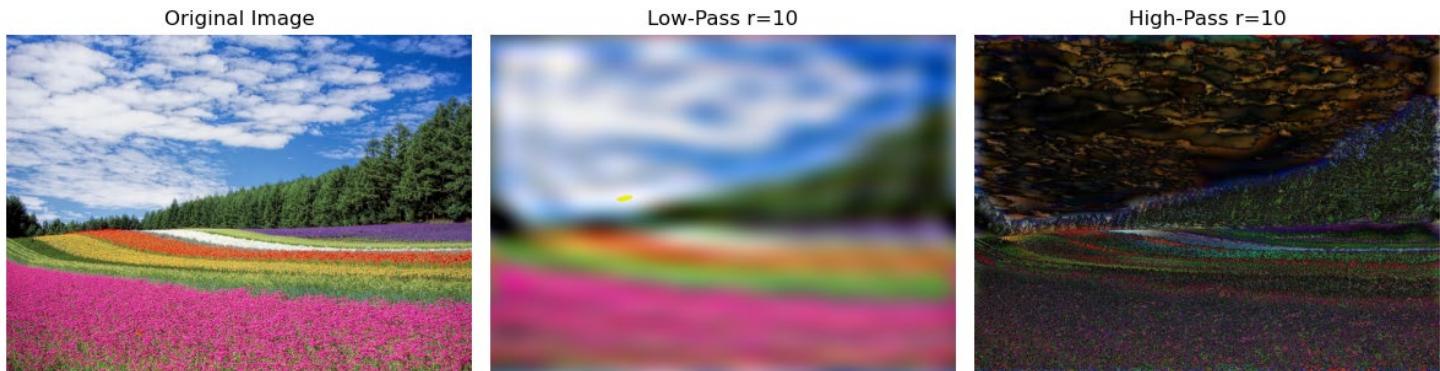
b. High-pass filter

$$H(u, v) = \begin{cases} 0, & \text{if } (u, v) \text{ is closer to the center than } r \\ 1, & \text{if } (u, v) \text{ is further from the center than } r \end{cases}$$

Code : [กดเพื่อไปดู code ที่ GitHub](#)

โดยที่ในกรณีของ High-Pass เนื่องจากภาพที่ได้ออกมาเราแทบจะมองไม่เห็นดังนั้น เราจะใช้เทคนิค CLAHE(Contrast Limited Adaptive Histogram Equalization) ในการปรับภาพให้มองเห็นได้ง่ายขึ้น

ภาพแรก flower1



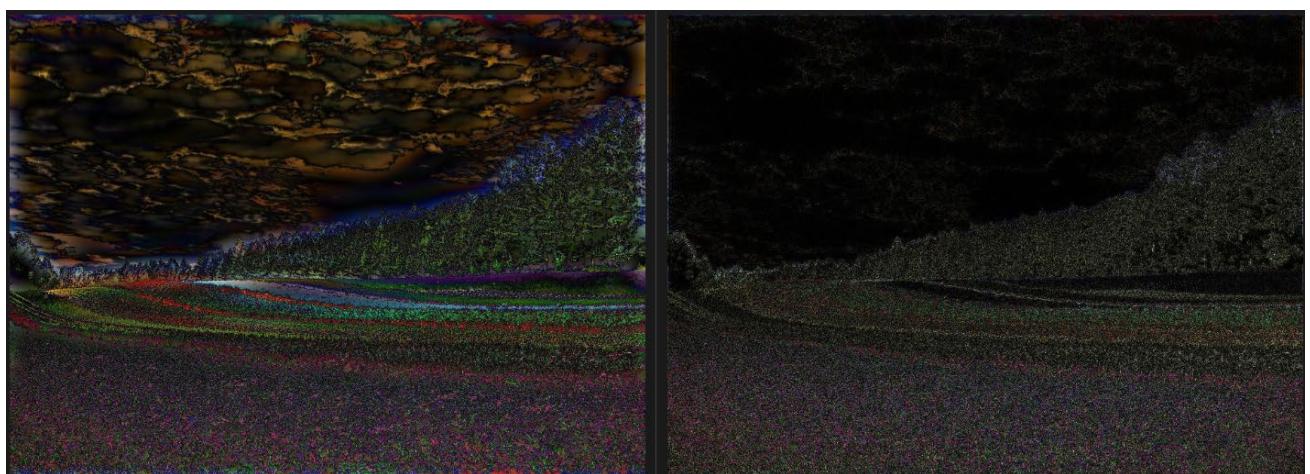
เริ่มจาก Low-Pass ก่อนจะเห็นว่าที่ $r=10$ หรือค่าที่มีค่าน้อย ซึ่งจะได้ภาพที่มีความเบลอมากที่สุดถ้าเทียบกับขนาด r อื่นๆ ทำให้สามารถบอกได้ว่าถ้ายิ่งค่า r มีค่าน้อย ภาพที่ได้ก็จะยิ่งมีความเบลอมากขึ้น

ทั้งนี้ก็เป็น เพราะว่า Low-Pass จะทำการกรองเอา high-frequency ออกทำให้ภาพออกมาเนื้อนกับการทำให้ภาพ smooth มากขึ้น (เพราะว่ามีแต่ low-frequency) จึงทำให้ภาพเบลอ ส่วนถ้า r มีค่ามากๆ เช่น $r=100$ ดังที่เห็นในภาพข้างบน จะได้ภาพที่มีความใกล้เคียงกับต้นฉบับ และที่ $r=50$ ถ้าลองเปรียบเทียบกับภาพต้นฉบับอีกครั้งจะเป็นดังนี้

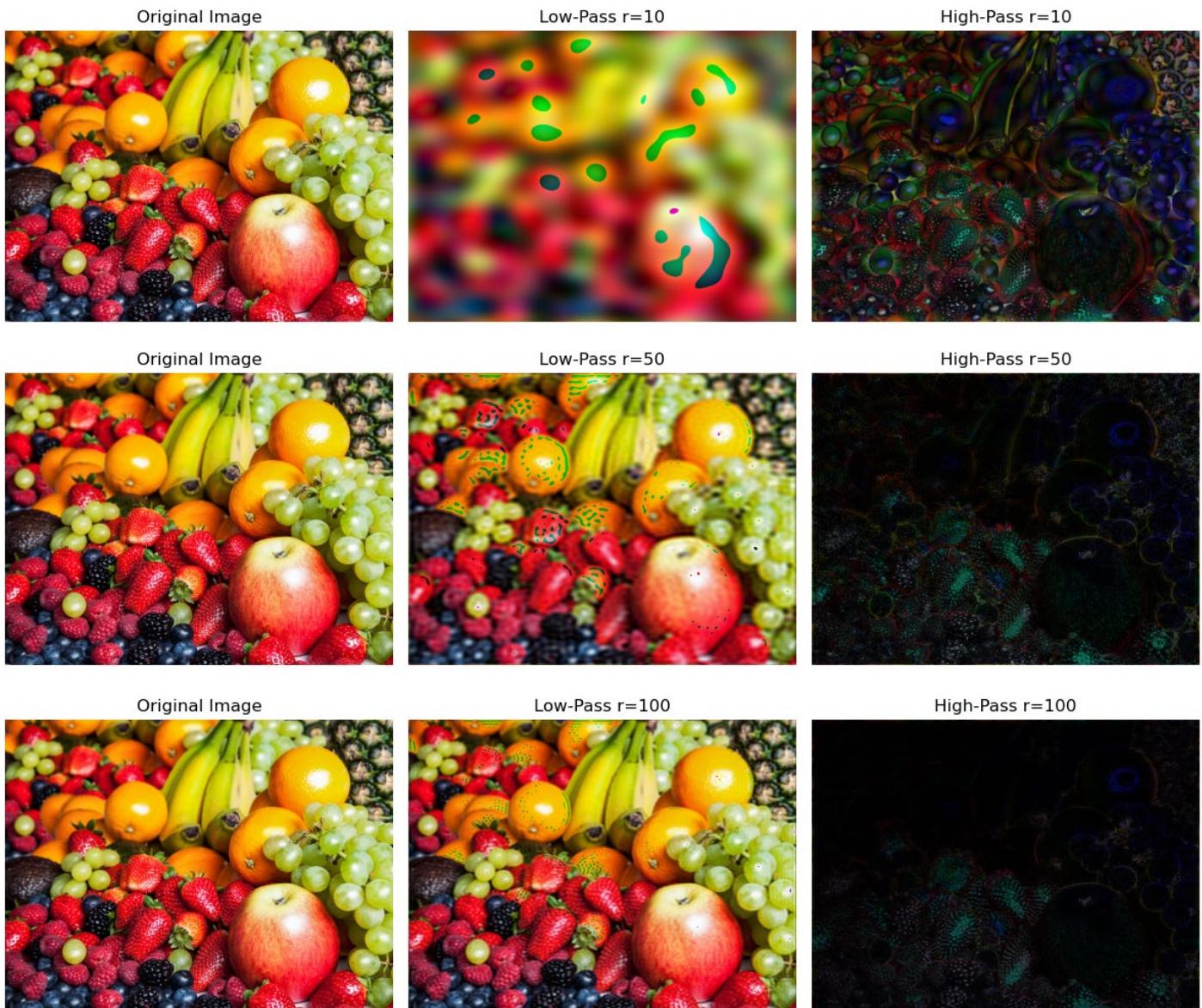


ซึ่งจะสังเกตเห็นว่าบริเวณส่วนของดอกไม้สีขาวมีส่วนของสีอื่นๆที่ไม่เกี่ยวข้องเข้ามาด้วยเกิดขึ้น ซึ่งอาจจะเป็น เพราะว่า เราใช้การกรองโดยการใช้จากการวัดรัศมีว่า ห่างจากจุดศูนย์กลางแค่ไหนทำให้ ในตอนที่ เราแยกสีเป็น 3 อันคือ RGB ก็อาจจะทำให้ แต่ละสีอาจมีส่วนที่เป็น high-frequency ที่ แตกต่างกันซึ่งทำให้ส่วนนั้นถูกตัดหายไปจึง เกิดเป็นสีแปลกๆออกมา และแน่นอนว่าหาก เราเพิ่ม r ให้มากขึ้นสีแปลกๆตรงนี้ก็จะน้อยลง (ปัญหานี้จะหายไปถ้าเราใช้ gaussian filter ซึ่งจะแสดงให้เห็น และ อธิบายว่า เพราะอะไรอย่างละเอียดในส่วนถัดไป)

ส่วน High-Pass จะตรงกันข้ามกับ Low-Pass คือ High-pass จะทำการกรองเอา low-frequency ออกทำให้เหลือแต่ high-frequency ก็คือ pixel ที่มี rate of change ที่สูง ซึ่งก็คือในส่วนของ edge โดยที่ เมื่อ $r=10$ หรือมีค่าที่น้อยจะได้ภาพที่ได้ จะมีรายละเอียดและขอบที่ละเอียดมาก ส่วนถ้า r มีค่า ก็จะคงภาพต้นฉบับไว้มากขึ้น ซึ่งหากเราทำการเปรียบเทียบที่ $r=10$ และ $r=100$ จะได้เป็นดังนี้



และอีกภาพคือ fruit ซึ่งได้ผลแบบเดียวกันกับ flower ดังนี้ โดยเราจะได้เห็นเหมือนกันว่า ใน Low-Pass เมื่อร มีค่าต่ำภาพที่ได้ก็จะคมabelelo แต่มีสีแปลกดูเกิดขึ้นดังที่ได้ตั้งข้อสังเกตไว้แล้วในภาพ flower และในส่วนของ high-pass ส่วนตัวคิดว่าเมื่อดูจาก flower กับ fruit แล้วพบว่าการใช้ค่า r น้อยๆในการตรวจจับ edge ของภาพดูจะมีความเหมาะสมที่สุด เพราะอย่างภาพผลไม้นี้ ถ้าดู High-pass ที่ $r=10$ จะเห็นว่าเรามองเห็นขอบของผลไม้ทุกชนิดในภาพ ในขณะที่ $r=50$ หรือ $r=100$ จะพบว่าภาพที่ได้เก็บขอบของผลไม้ไม่ได้ทั้งหมด



โดยภาพทั้งหมดสามารถ download หรือดูได้ที่ : [GitHubQ1-OUTPUT](#)

2. Gaussian filters

a. Low-pass filter

$$H(u, v) = e^{-D^2(u,v)/2D_0^2}$$

b. High-pass filter

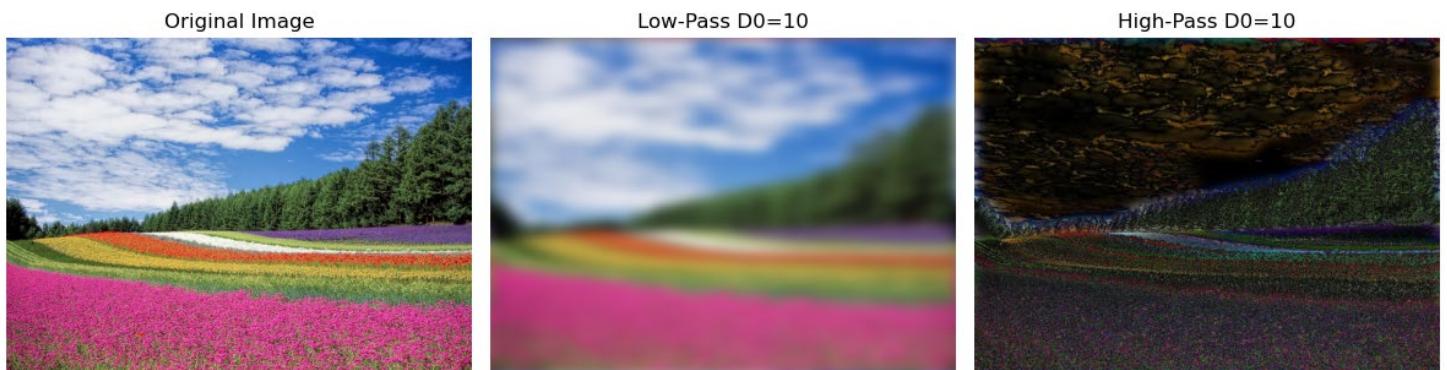
$$H(u, v) = 1 - e^{-D^2(u,v)/2D_0^2}$$

Code : [กดเพื่อไปดู code ที่ GitHub](#)

ผลที่ได้เป็นดังนี้

เช่นเดียวกับ notch filters ในกรณีของ High-Pass เนื่องจากภาพที่ได้ออกมาเราแทนจะมองไม่เห็นดังนั้น เราจะใช้เทคนิค CLAHE(Contrast Limited Adaptive Histogram Equalization) ในการปรับภาพสว่างขึ้นทำให้มองเห็นได้ง่ายขึ้น

ภาพแรก flower1



เริ่มจาก low-pass จะเห็นว่า Gaussian filter ให้ผลที่คล้ายๆ กับ Notch filter แต่จะสังเกตว่าบริเวณ

Original Image

Low-Pass D0=100

High-Pass D0=100



ทุ่งดอกไม้สีขาวที่เราได้ให้ข้อสังเกตไปใน notch filter ว่ามีสีแปลกลๆ โผล่ออกมาบ้าง ตอนนี้ได้หายไปแล้ว

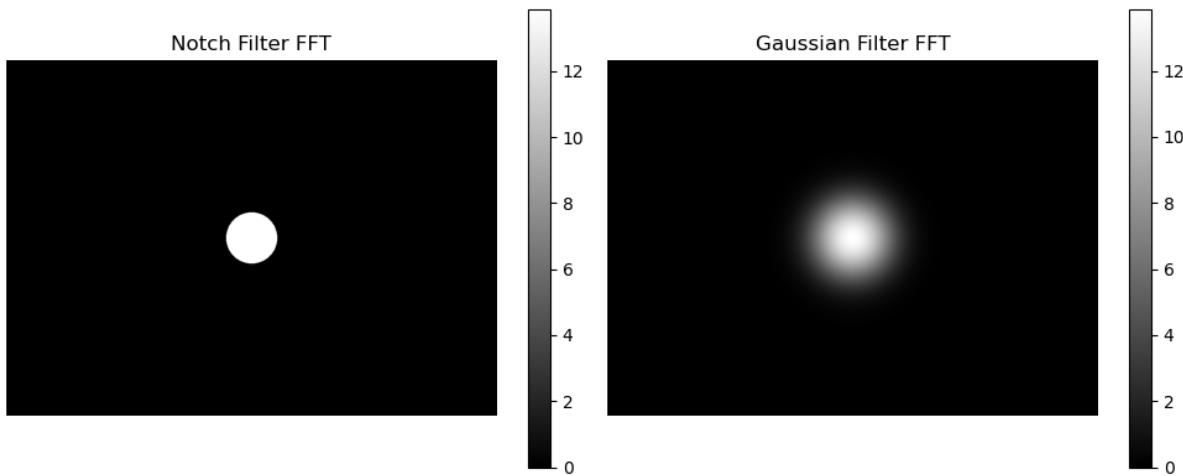
โดยถ้าเราเปรียบเทียบ Gaussian filter ที่ $D=50$ กับ Notch filter ที่ $r=50$ จะได้ดังนี้ ภาพซ้ายจะเป็นของ Gaussian และด้านขวาจะเป็น Notch



จะสังเกตว่าตอนนี้สีเปลกๆได้หายไปแล้ว

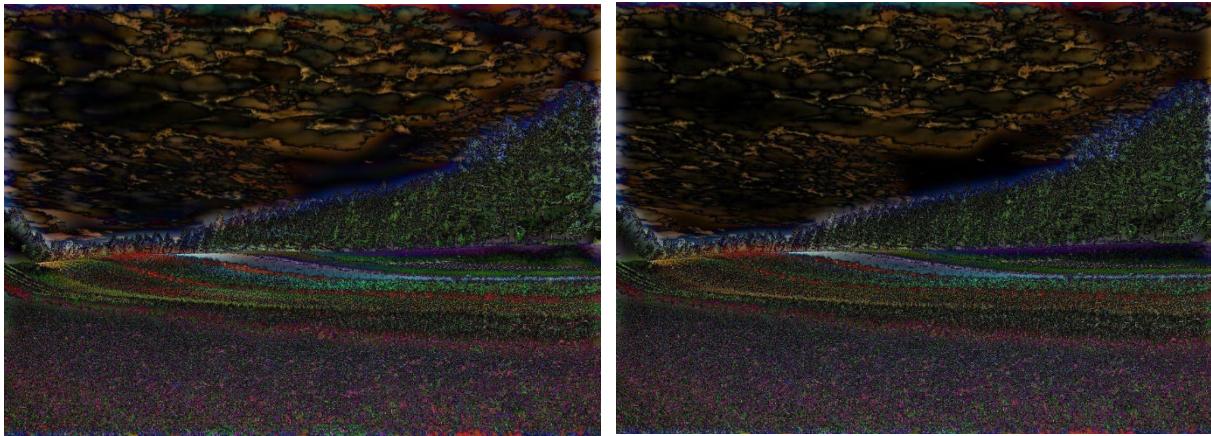


ที่เป็นเช่นนี้เนื่องจากหากเราลองทำการดู Fourier Transform ของfilter ที่ได้จากของ notch และ gaussian filter ของทั้งสี R,G,B (เนื่องจากทั้งหมดคล้ายกันมากจึงขอแสดงแค่ของ R อย่างเดียว)



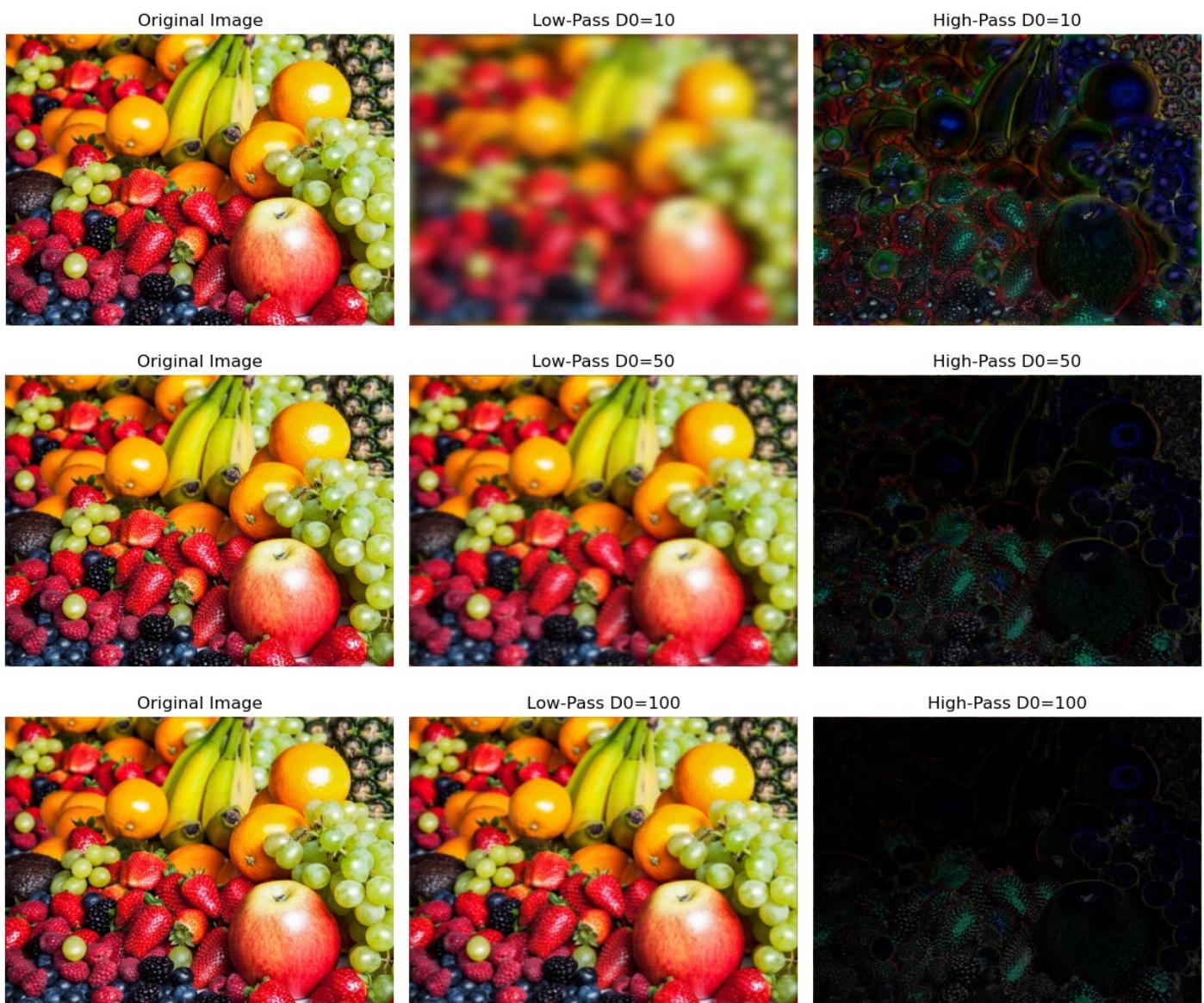
จะเห็นว่า Gaussian Filter มีเหมือนกับการค่อยๆลดระดับลงไปจนขาวสุดที่ตรงกลาง ในขณะที่แบบ Notch Filter นั้นเป็นสีขาวทั้งหมด ทำให้เมื่อเราใช้ Gaussian Filter จึงได้ภาพออกมาเป็นสีของทั้ง R,G,B ที่ผสมกันที่ไม่ถูกตัดออกไปทั้งหมดแบบ Notch ทำให้ภาพที่เราได้ไม่เกิดสีเปลกๆเกิดขึ้น

ในส่วนของ high-pass ถ้าเราลองเอาของทั้ง Notch(ช้าย) และ Gaussian(ขวา) ที่ $r=10$ และ $D0=10$ มาเปรียบเทียบกันจะได้ผลดังนี้



จะเห็นว่าผลลัพธ์ที่ได้ไม่ต่างกันเลย (ดูด้วยตาเปล่า) ซึ่งเราจะไปดูความแตกต่างของสองแบบนี้ก็ต้องในข้อที่ 3 โดยในข้อที่ 3 เราจะทำการบวกเข้ากับภาพต้นฉบับด้วย จะได้เห็นว่าภาพคอมซัดขึ้นหรือไม่

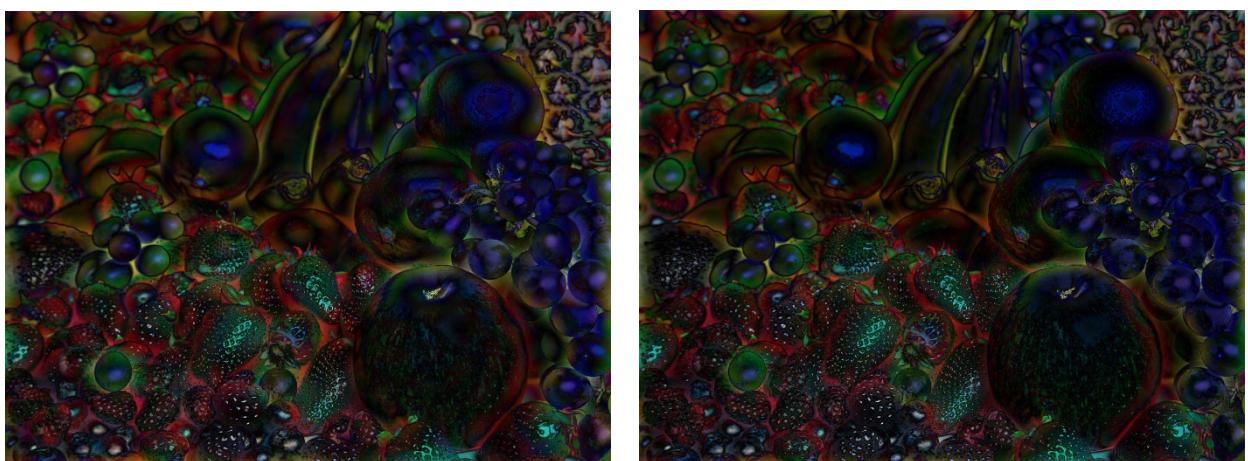
ภาพที่สอง fruit ได้ผลเช่นเดียวกันกับภาพ flower ซึ่งจะเห็นว่าตอนนี้ใน low-pass ไม่มีสีเจียวແປลากๆ อยู่บนผลไม้แล้ว (ลองเปรียบเทียบ notch จะเห็นว่ามีสีແປลากๆ เกิดขึ้นบนผิวของผลไม้ ดูภาพด้านล่าง)



สองภาพด้านล่างจะเป็นการเปรียบเทียบ Low-Pass ของ Notch(ซ้าย) กับ Gaussian(ขวา) ที่ $r=50$ และ $D=50$

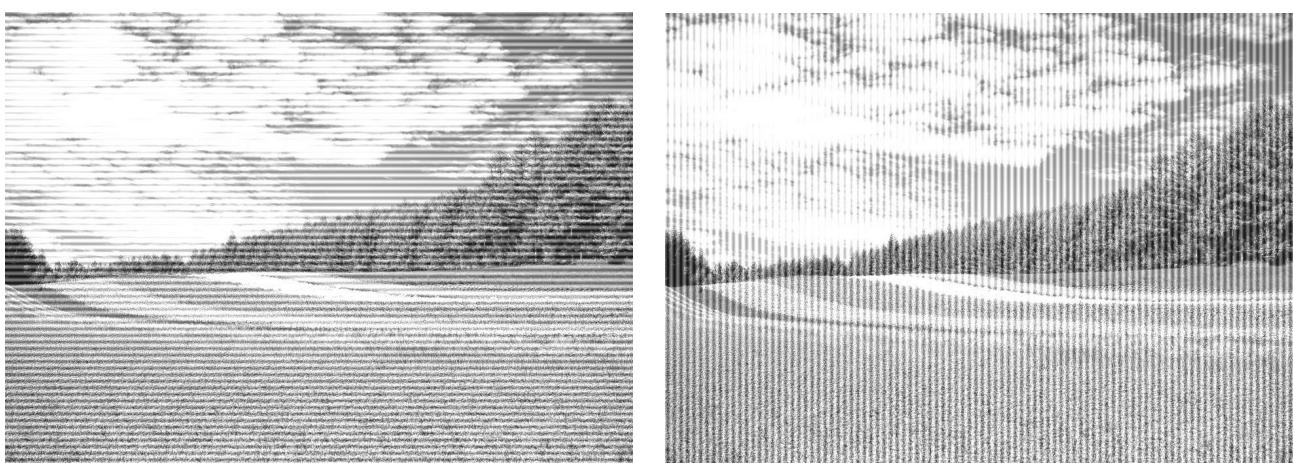


ในส่วนของ high pass ก็เช่นเดียวกันกับ high pass ของ notch โดยที่เราลองเปรียบเทียบทั้งสองfilterก็จะพบว่าภาพที่ได้ไม่ต่างกันเลยถ้ามองด้วยตาเปล่า (ด้านซ้ายของnotchด้านขวาของgaussian)



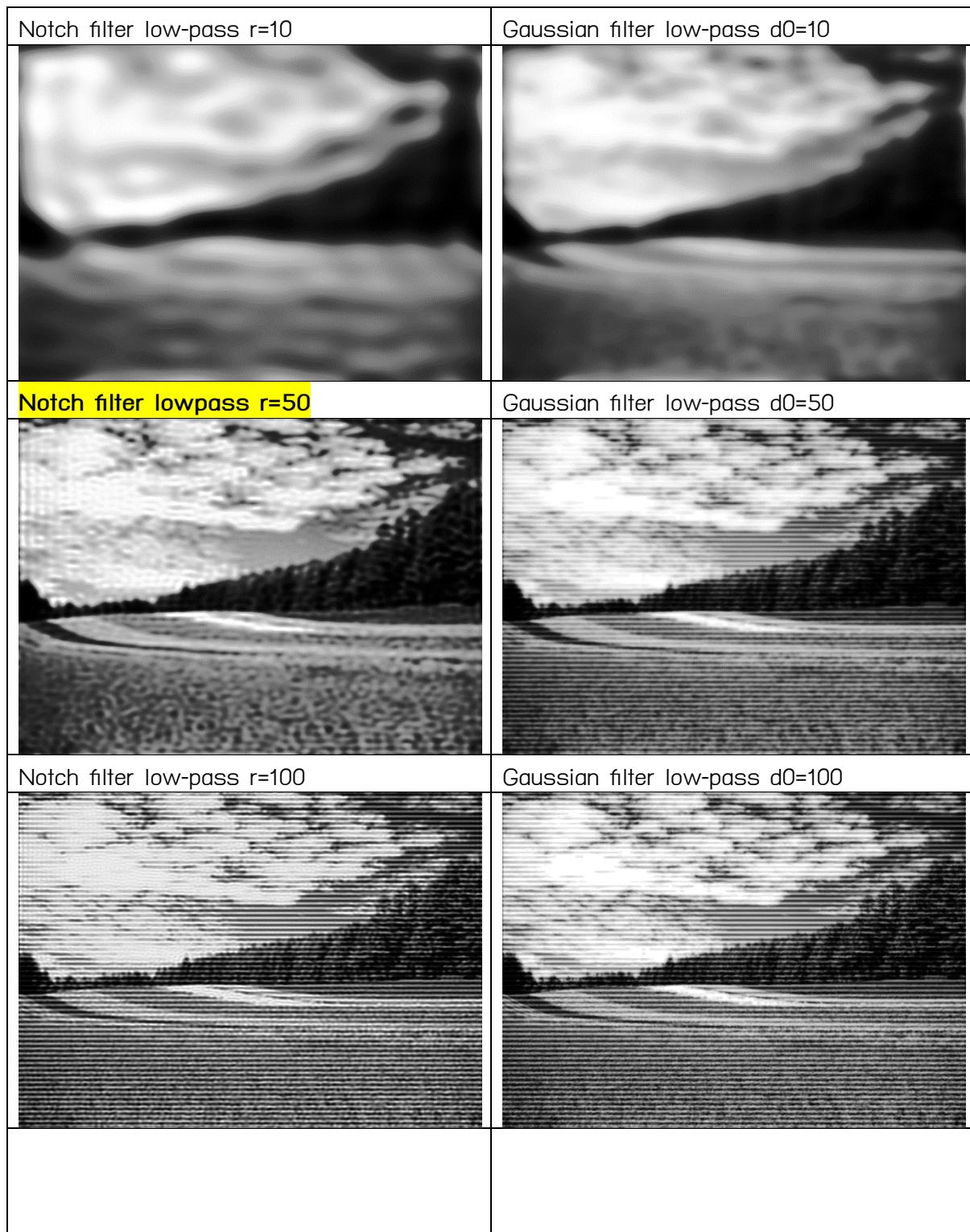
โดยภาพทั้งหมดสามารถdownloadหรือดูได้ที่ : [GitHubQ2-OUTPUT](#)

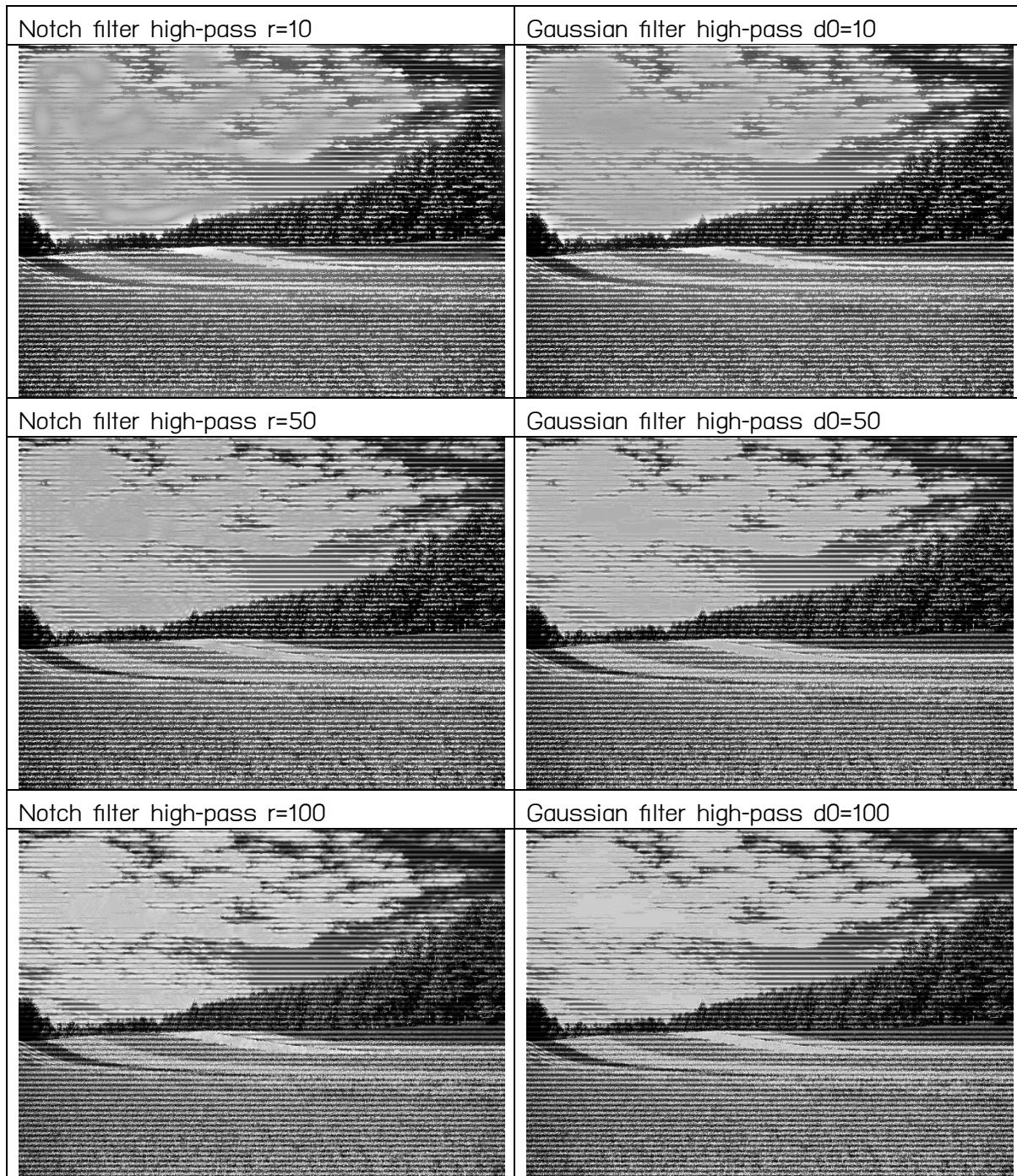
3. Remove periodic noise from the given images using any of the filters that you think are the most suitable ones to be used. Code : [กดเพื่อไปดูcodeที่GitHub](#)



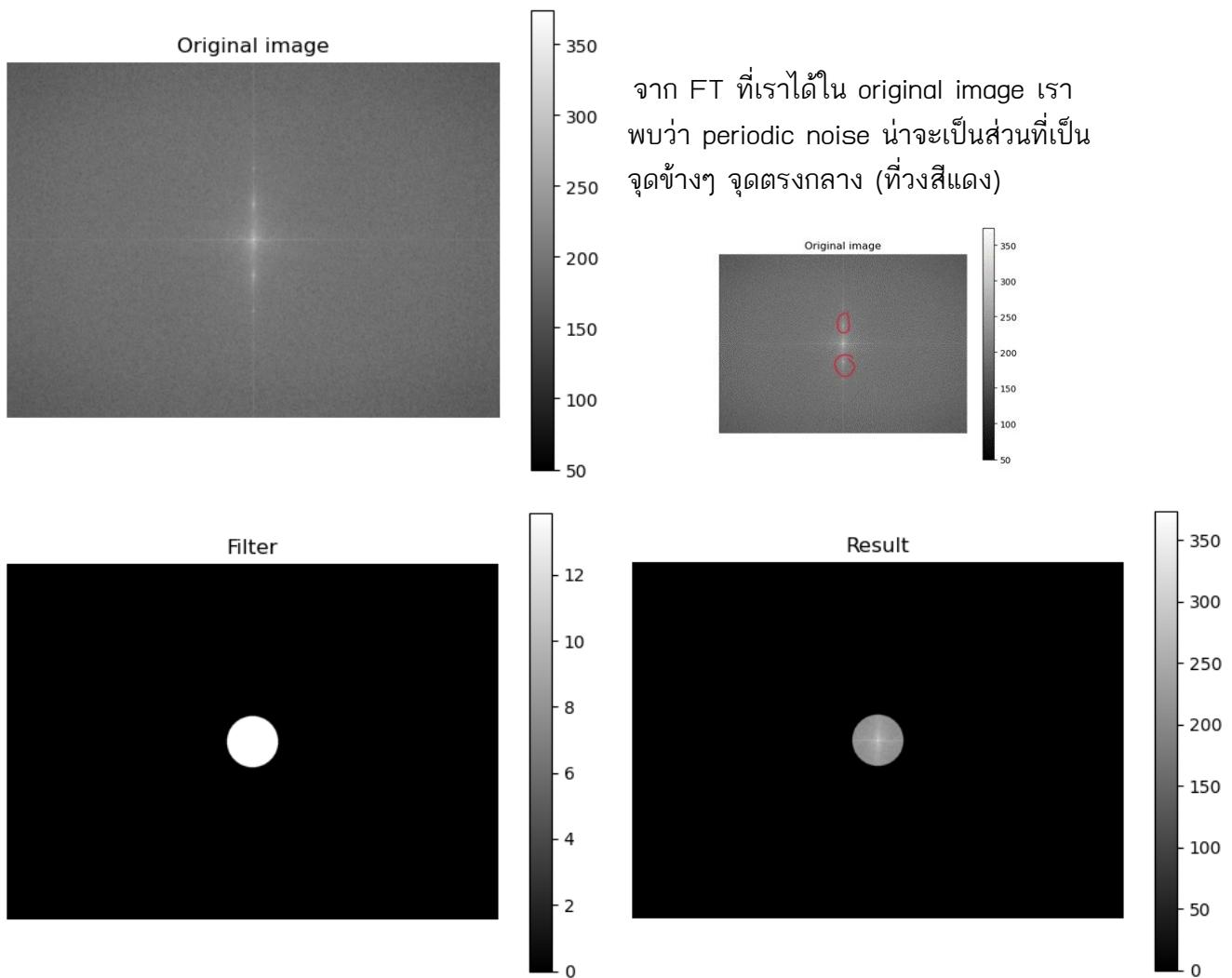
จากภาพจะพบว่าเรามี periodic ส่องแบบคือ แนวตั้งกับแนวนอน โดยส่วนตัวคิดว่าจะต้องใช้แบบที่เป็น low-pass เพื่อที่จะได้พยากรณ์เกลี่ยสิ หรือทำให้ภาพ smooth ขึ้นจนทำให้ periodic noise หายไปแต่เพื่อการตัดสินใจที่ง่ายขึ้น เราจะทำการลองทุกแบบ คือ ใช้ notch filter และ gaussian filter ทั้งแบบ low-pass และ high-pass โดยที่ทั้งสองภาพเพื่อที่เราจะได้แยกความแตกต่างได้ง่ายขึ้น จึงทำการใช้เทคนิคของ Local histogram equalization เพื่อดึง contrast ของภาพให้ออกมาเหมือนกัน และในส่วนของ high-pass เราได้ทำการบวกเข้าไปกับภาพต้นฉบับด้วย (เพราะว่า high-pass เพียงอย่างเดียวเราได้แค่ขอบ ของภาพต้นฉบับ) ซึ่งได้ผลดังนี้

เริ่มจาก periodic noise แบบที่เป็นแนวนอน หรือก็คือภาพ Noisy_flower1_horizontal



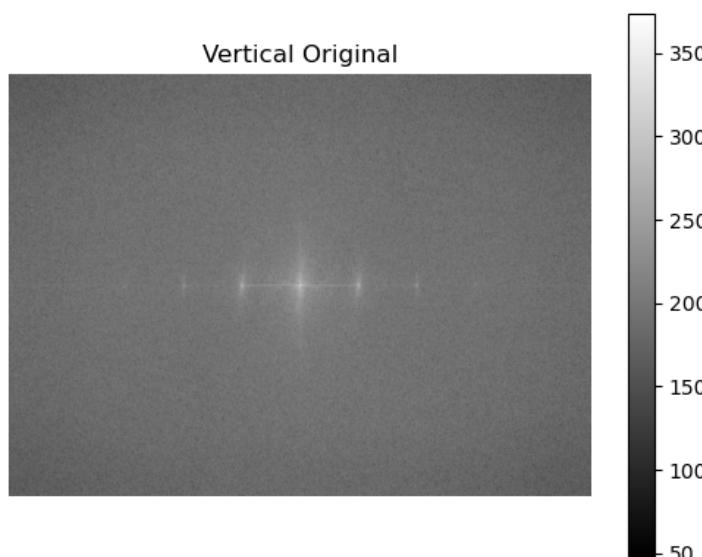


จะเห็นว่า high-pass ทำให้ภาพมีความคมมากขึ้นกว่าเดิมซึ่งไม่เหมาะสมในการลบ periodic noise ดังนั้น low-pass จึงจะดูเหมาะสมที่สุด โดยจากตารางที่เราเปรียบเทียบไปจะพบว่า Notch filter แบบ low-pass จากค่า r ทั้งสามค่า มีค่า $r=50$ ที่ดีที่สุดที่ลบ noise ออกໄไปได้ และยังคงให้ภาพที่ไม่เบลอ จนเกินไปจนมองไม่ออกว่าเป็นภาพอะไรเหมือนที่ $r=10$ ถ้าเราลองดู FFT ของค่าตอบนี้จะเป็นดังนี้

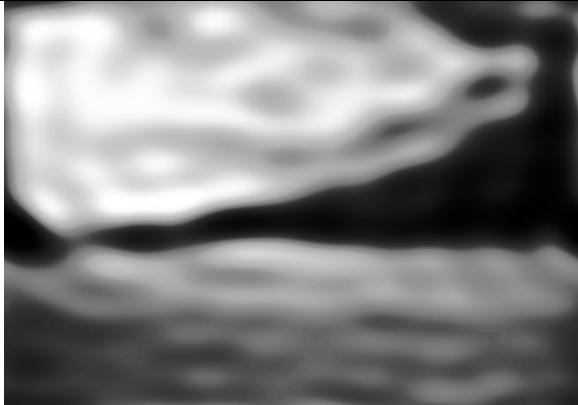
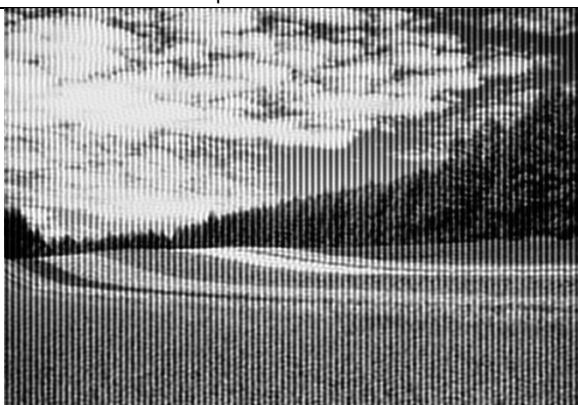
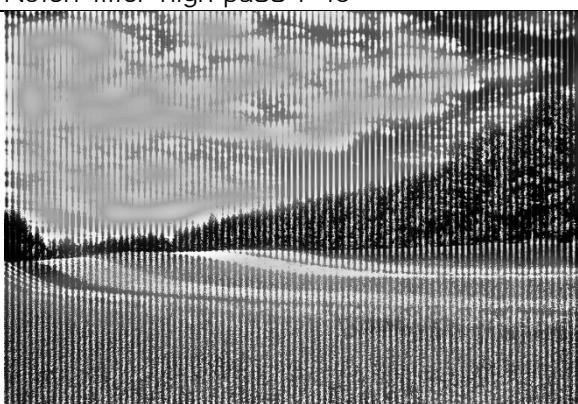


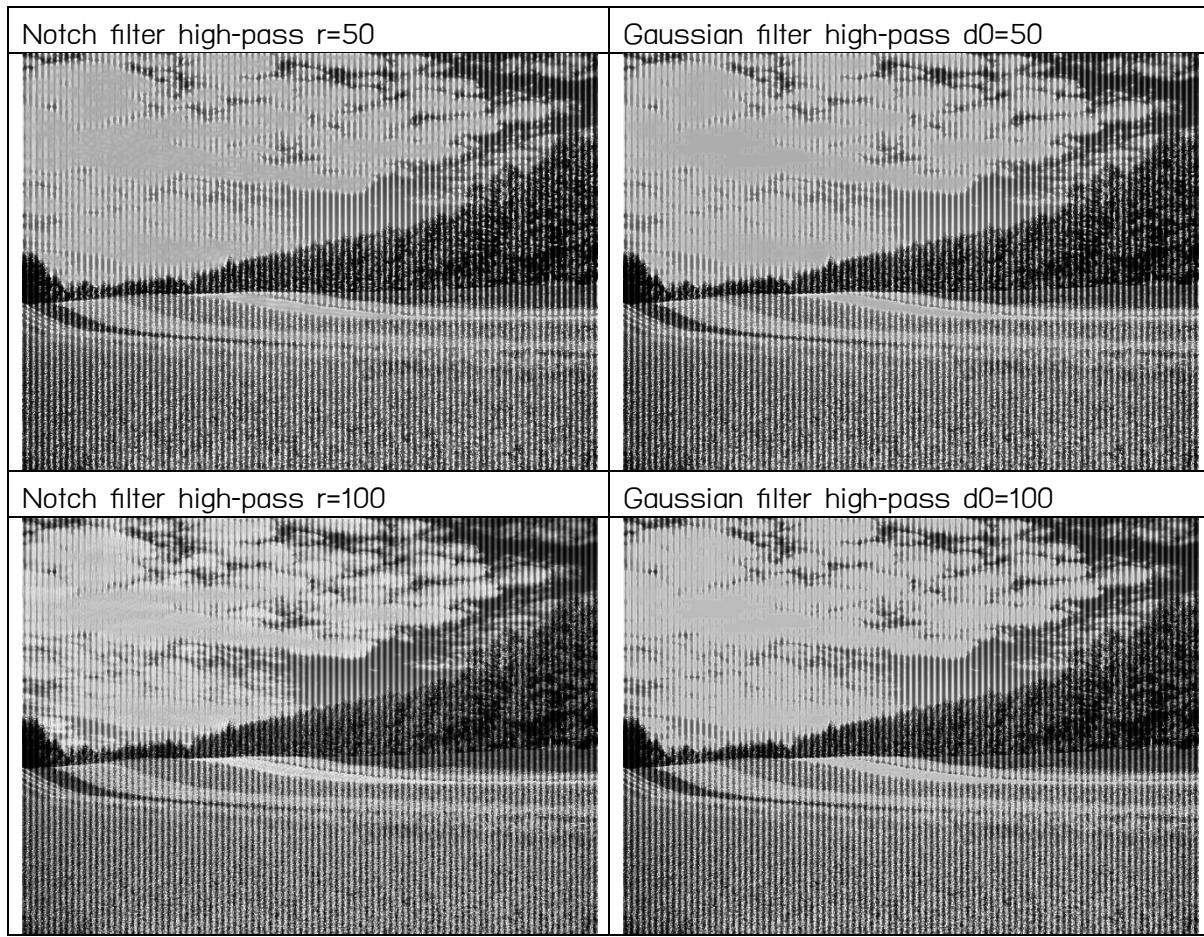
ทำให้พอเรารายใช้ notch filter ขนาด $r=50$ จึงทำการคุณส่วนเฉพาะที่เป็นภาพที่ไม่มี periodic noise ทำให้การใช้ $r=50$ จะลบ periodic noise ออกໄไปได้

ต่อมาเป็นภาพ periodic noise ในแนวตั้งบ้าง (Noisy_flower1_vertical) ซึ่งก็คาดว่าเราจะใช้ notch filter ที่ขนาด $r=50$ เช่นกันเนื่องจาก Original image มี FT เป็นดังนี้



โดยถ้าเราจะเปรียบเทียบก็จะได้ผลคล้ายๆกันกับภาพแรกที่เป็น periodic noise ในแนวนอน

Notch filter low-pass r=10	Gaussian filter low-pass d0=10
	
Notch filter lowpass r=50	Gaussian filter low-pass d0=50
	
Notch filter low-pass r=100	Gaussian filter low-pass d0=100
	
Notch filter high-pass r=10	Gaussian filter high-pass d0=10
	



ซึ่งเป็นไปตามที่คาดไว้ว่าการใช้ Notch filter แบบ high-pass โดยใช้ $r=50$ จะลบ periodic noise ได้ดีที่สุด

โดยภาพทั้งหมดสามารถ download หรืออ่านได้ที่ : [GitHubQ3-OUTPUT](#)