โปรแกรมคอมพิวเตอร์เพื่อตรวจคัดกรองโรค Vestibular Dysfunction

Screening Computer Software for Vestibular Dysfunction

ธนพล วงศ์วราพุทธ

Thanapol Wongwaraputh

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต

สาขาวิศวกรรมชีวการแพทย์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2560

โปรแกรมคอมพิวเตอร์เพื่อตรวจคัดกรองโรค Vestibular Dysfunction

Screening Computer Software for Vestibular Dysfunction

นายธนพล วงศ์วราพุทธ รหัสประจำตัวนักศึกษา 57010550

อาจารย์ที่ปรึกษา

รศ.ดร.ชูชาติ ปิณฑวิรุจน์

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต

สาขาวิศวกรรมชีวการแพทย์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2560

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2560

สาขาวิชา วิศวกรรมชีวการแพทย์

คณะวิศวกรรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง โปรแกรมคอมพิวเตอร์เพื่อตรวจคัดกรองโรค Vestibular Dysfunction

Screening Computer Software for Vestibular Dysfunction

ผู้จัดทำ

นายธนพล วงศ์วราพุทธ รหัสประจำตัวนักศึกษา 57010550

ปริญญานิพนธ์นี้ผ่านการตรวจสอบโดยอาจารย์ที่ปรึกษาแล้ว

(รศ.ดร.ชูชาติ ปิณฑวิรุจน์)

อาจารย์ที่ปรึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อโครงงาน โปรแกรมคอมพิวเตอร์เพื่อตรวจคัดกรองโรค Vestibular Dysfunction

นักศึกษา นายธนพล วงศ์วราพุทธ **รหัสนักศึกษา** 57010550

ปริญญา วิศวกรรมศาสตรบัณฑิต

สาขาวิชา วิศวกรรมชีวการแพทย์

ปีการศึกษา 2560

อาจารย์ที่ปรึกษา รศ.ดร. ชูชาติ ปิณฑวิรุจน์

บทคัดย่อ

โรค Vestibular Dysfunction เป็นหนึ่งในสาเหตุที่ก่อให้เกิดอาการวิงเวียนศีรษะแบบบ้านหมุน ในปัจจุบันโรคนี้ไม่ได้เป็นที่รู้จักอย่างกว้างขวางแม้ในวงการแพทย์ของไทย การวินิจฉัยโรคต้องใช้แพทย์ผู้เชี่ยวชาญและมีประสบการณ์โดยดูระยะเวลาของการกรอกตา การใช้โปรแกรมคอมพิวเตอร์เข้ามาช่วยจึงมีประโยชน์ในการสนับสนุนการวินิจฉัยโรคของแพทย์ โปรแกรมที่สร้างขึ้นทำงานโดยใช้วิดีโอบันทึกขณะแพทย์ตรวจผู้ป่วยแล้วนำมาตรวจจับระยะเวลาการกรอกตา และแสดงเป็นกราฟการตำแหน่งของตาดำ ผลลัพธ์ของโปรแกรมค่อนข้างเป็นที่น่าพอใจแต่ต้องมีการพัฒนาปรับปรุงอีกในหลายด้านและในอนาคตจะนำไปทดสอบกับผู้ป่วยจริง และพัฒนาเพื่อสามารถทำงานได้ในสมาร์ทโฟนต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Project Title	Screening Computer Software for Vestibular Dysfunction
Student	Thanapol Wongwaraputh Student ID 57010550
Degree	Bachelor of Engineering
Program	Biomedical Engineering
Year	2018
Advisor	Assoc. Prof. Dr. Chuchart Pintavirooj

Abstract

This project is recommended by a doctor, form SWU. According to the Vestibular dysfunction, one of the reasons cause vertigo, is not wildly known even medical society of Thailand. There is a few of doctors that have experience about the disease and diagnosis. The diagnosis can be wrong due to a fatigue of a doctor's eye muscles. So using a computer software is beneficial for both doctor for supporting the diagnosis and the patients who want to screen themselves before seeing doctor. Moreover this project also provides the information of disease and the diagnosis. This project is creating software that uses a doctor's diagnosis video to detect eyes and calculate duration of eye movement then save and show result that a patient has a disorder or not.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

วิทยานิพนธ์เล่มนี้สำเร็จได้ด้วยความกรุณาจากบุพการีอันประกอบด้วยบิดา มารดาผู้ให้โอกาสทางการศึกษาและกำลังใจกับผู้จัดทำเสมอมาจนโครงงานนี้ประสบความสำเร็จ ขอบคุณ ผศ. นพ.วิศาล มหาสิทธิวัฒน์ ซึ่งเป็นผู้แนะนำงานวิจัยนี้ พร้อมทั้งให้คำแนะนำให้ข้อมูลเกี่ยวกับโรคฯ การวินิจฉัยอีกทั้งยังให้คำปรึกษาในหลายๆด้าน และขอบคุณ อาจารย์ที่ปรึกษา รศ.ดร.ชูชาติ ปิณฑวิรุจน์ ที่ให้คำชี้แนะช่วยแก้ปัญหาตลอดจนให้ความรู้พร้อมสนับสนุนให้มีประสบการณ์ที่ดีสามารถนำไปพัฒนางานวิจัยเพื่อให้เกิดคุณประโยชน์ต่อผู้อื่นและประเทศชาติในอนาคตต่อไป ขอบพระคุณครอบครัวของผู้จัดทำที่พร้อมช่วยเหลือในทุกด้านและให้กำลังใจเกี่ยวกับการเรียน การทำโครงงานทำให้วิทยานิพนธ์เล่มนี้สำเร็จได้ด้วยดี ขอบคุณเพื่อนๆและพี่ปริญญาโท สาขาวิศวกรรมชีวการแพทย์ที่ให้คำปรึกษาและช่วยเหลือในทุกด้านโดยตลอดเมื่อมีปัญหาพร้อมให้คำแนะนำต่างๆจากประสบการณ์การทำงานวิจัยทำให้ปริญญานิพนธ์นี้สามารถสำเร็จไปได้ด้วยดี สำหรับคุณงามความดีที่เกิดจากปริญญานิพนธ์ฉบับนี้ ข้าพเจ้าขอมอบให้กับบิดามารดาที่รักและเคารพ ตลอดจนครูบาอาจารย์ที่เคารพทุกท่านที่ได้ประสิทธิ์ประสาทวิชาความรู้และถ่ายทอดประสบการณ์ที่ดีให้แก่ข้าพเจ้า

ธนพล วงศ์วราพุทธ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทคัดย่อ I

บทคัดย่อภาษาอังกฤษ II

กิตติกรรมประกาศ III

สารบัญ IV

สารบัญรูป VI

บทที่ 1 บทนำ 1

1.1 ความเป็นมาและความสำคัญของปัญหา เพิ่มเลข Ref 1

1.2 วัตถุประสงค์ของการศึกษา 1

1.3 ขอบเขตการวิจัย 1

1.4 ขั้นตอนการศึกษา 2

บทที่ 2 เอกสารที่เกี่ยวข้อง 3

2.1 Vestibular system 3

2.1.1 ระบบหลอดกึ่งวงกลม (semicircular canal system) 4

2.1.2 ทำงานเป็นระบบดันและดึง 4

2.1.3 Vestibulo-Ocular Reflex (VOR) 5

2.1.4 Head Impulse Test (HIT) สำหรับ VOR 6

2.2 การทรงตัว 8

2.3 อาการเวียนศีรษะ 11

2.4 Python Programming Language 13

2.5 OpenCV 16

2.5.1 Haar Cascade Classifier 16

2.5.2 Template matching 18

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6 Dlib library 19

บทที่ 3 วิธีการดำเนินงานวิจัย 21

3.1 การออกแบบโปรแกรม 21

3.2 กรอบแนวคิดในการทำวิจัย 24

3.3 การทดสอบโปรแกรม 24

บทที่ 4 ผลการทดลอง 25

4.1 ผลการทดลอง 25

บทที่ 5 สรุปผลและปัญหาที่เกิดขึ้น 27

5.1 สรุปผลและอภิปรายผล 27

5.2 ปัญหาและแนวทางแก้ไข 28

เอกสารอ้างอิง 29

ภาคผนวก 31

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

หน้า

รูปที่ 2.1 หูชั้นใน 3
รูปที่ 2.2 ระบบดันและดึงของหลอดกึ่งวงกลม 5
รูปที่ 2.3 แสดงการทำงานของ vestibulo-ocular reflex 6
รูปที่ 2.4 Head Impulse Test เพื่อทดสอบ VOR 7
รูปที่ 2.5 กราฟความเร็วของศีรษะ และดวงตา ขณะทำ Head Impulse Test 7
รูปที่ 2.6 ไฟล์ Haar Cascade Classifier ทั้งหมดจาก OpenCV 17
รูปที่ 2.7 ตัวอย่างการตรวจจับใบหน้าด้วย Haar Cascade Classifer 18
รูปที่ 2.8 การทำ template matching 19
รูปที่ 2.9 ตัวอย่างการใช้ Dlib เพื่อหา facial landmark 68 จุด 20
รูปที่ 3.1 หน้าต่างเริ่มต้นของโปรแกรมฯ 21
รูปที่ 3.2 ROI บริเวณดวงตา และจุดอ้างอิงที่ใช้ในการคำนวณ 22
รูปที่ 3.3 แผนภาพการทำงานของโปรแกรม 23
รูปที่ 4.1 โปรแกรมขณะทำการตรวจหาใบหน้าและตาดำ 25
รูปที่ 4.3 กราฟตำแหน่งของตาดำซ้ายและขวา 26
รูปที่ 4.4 หน้าต่างของโปรแกรมฯเมื่อกด show graph 26

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

โครงงานนี้ได้รับการเสนอแนะจากแพทย์ เนื่องจากโรค Vestibular Dysfunction ซึ่งเป็นหนึ่งในสาเหตุที่ทำให้เกิดอาการเวียนศีรษะหรือบ้านหมุนนั้นไม่ได้ถูกรู้จักอย่างกว้างขวางแม้ในวงการแพทย์ของไทย การวินิจฉัยโรคต้องใช้แพทย์ผู้มีประสบการณ์และรู้จักโรคนี้ซึ่งมีอยู่ไม่มาก อีกทั้งการวินิจฉัยโรคอาจผิดพลาดได้หากแพทย์มีอาการเมื่อยล้าในการใช้สายตา ดังนั้นการใช้โปรแกรมคอมพิวเตอร์เข้ามาช่วยจึงมีผลดีทั้งต่อแพทย์ในการสนับสนุนการวินิจฉัยและผู้ป่วยในการตรวจคัดกรองโรคก่อนพบแพทย์ อีกทั้งยังเป็นการเผยแพร่ข้อมูลงานวิจัยเกี่ยวกับโรคฯและการวินิจฉัยโรคอีกด้วย

1.2 วัตถุประสงค์ของการศึกษา

1. เพื่อศึกษาการเขียนโปรแกรมด้วยภาษา Python
2. เพื่อศึกษาการทำงานของโปรแกรมตรวจคัดกรองโรค Vestibular Dysfunction
3. เพื่อสร้างโปรแกรมสนับสนุนการวินิจฉัยโรค Vestibular Dysfunction ของแพทย์
4. เพื่อสร้างโปรแกรมตรวจคัดกรองโรค Vestibular Dysfunction ที่สามารถตรวจได้เองก่อนมาพบแพทย์
5. เพื่อพัฒนาต่อยอดโปรแกรมตรวจคัดกรองโรคในสมาร์ทโฟน

1.3 ขอบเขตการวิจัย

งานวิจัยนี้ได้นำเสนอการออกแบบและพัฒนาโปรแกรมคอมพิวเตอร์เพื่อการตรวจคัดกรองโรค Vestibular Dysfunction โดยมีขอบเขตงานวิจัยต่อไปนี้

คุณสมบัติส่วนของซอฟต์แวร์

1.3.2.1 ใช้ภาษา Python, library OpenCV และ Dlib เพื่อสร้างโปรแกรมประมวลผลภาพและวิเคราะห์เพื่อการตรวจคัดกรองโรค Vestibular Dysfunction

1.3.2.2 โปรแกรมใช้ทดสอบกับวิดีโอ ซึ่งควรมีความละเอียด และ frame rate ที่สูง ในเบื้องต้นคาดว่าควรมีความละเอียด 1280x720, 120 frames/second

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.4 ขั้นตอนการศึกษา

ในการศึกษาและวิจัยในครั้งนี้ได้แบ่งขั้นตอนของการศึกษาได้ดังต่อไปนี้

1.4.1 ศึกษาโรค Vestibular Dysfunction และวิธีการวินิจฉัยโรคจากแพทย์

1.4.2 ศึกษาภาษา Python และ OpenCV

1.4.3 ศึกษาการเขียนโปรแกรมเพื่อประมวนผลภาพ

1.4.4 ออกแบบโปรแกรมตรวจคัดกรองโรค Vestibular Dysfunction

1.4.5 เขียนโปรแกรมตรวจคัดกรองโรค Vestibular Dysfunction

1.4.6 ทดสอบและบันทึกผลการทดลอง

1.4.7 หาข้อผิดพลาดของโปรแกรม

1.4.8 ทดสอบโปรแกรมด้วยวิดีโอของผู้ทดสอบที่สุขภาพดี บันทึกผลการทดลองและหาข้อผิดพลาดของโปรแกรมพร้อมปรับปรุงแก้ไข

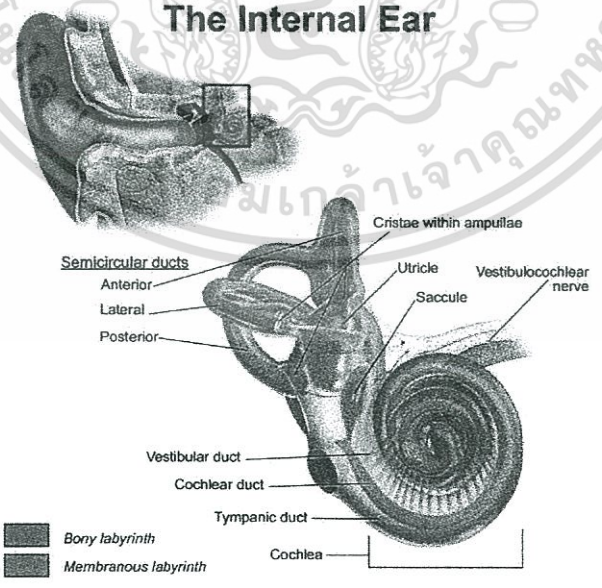
1.4.9 สรุปผลการทดลอง รวบรวมปัญหาและข้อเสนอแนะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
เอกสารที่เกี่ยวข้อง

2.1 Vestibular system

Vestibular system [1] เป็นระบบที่มีส่วนในการรักษาดุล (balance) ในสัตว์เลี้ยงลูกด้วยนมโดยมาก และในความรู้สึกเกี่ยวข้องกับทิศทางของร่างกายในปริภูมิ (spatial orientation) เป็นระบบรับความรู้สึกที่ให้ข้อมูลสำคัญของร่างกายเกี่ยวกับการเคลื่อนไหวและความรู้สึกเกี่ยวกับดุล (equilibrioception หรือ sense of balance). ในสัตว์เลี้ยงลูกด้วยนมโดยมาก vestibular system พร้อมกับหูชั้นในรูปหอยโข่ง (cochlea) ซึ่งเป็นอวัยวะในระบบการได้ยิน ประกอบกันเป็นห้องหูชั้นใน (labyrinth of the inner ear) อยู่ในช่องของหูชั้นใน เพราะว่าการเคลื่อนไหวมีทั้งแบบหมุนและแบบเลื่อน vestibular system ก็มีส่วนประกอบสองอย่างเหมือนกัน คือ ระบบหลอดกึ่งวงกลม (semicircular canal) ซึ่งบอกการเคลื่อนไหวแบบหมุน และระบบ otoliths ซึ่งบอกความเร่งแนวเส้น vestibular system โดยหลักส่งข้อมูลไปยังโครงสร้างประสาทที่ควบคุมการเคลื่อนไหวของตา เป็นฐานทางกายภาพสำหรับ vestibulo-ocular reflex ซึ่งจำเป็นในการเห็นที่ชัดเจน, และไปยังกล้ามเนื้อที่ควบคุมอากัปกิริยาเพื่อให้สัตว์สามารถทรงตัวไว้ได้



รูปที่ 2.1 หูชั้นใน

ที่มา https://en.wikipedia.org/wiki/Inner_ear

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.1 ระบบหลอดกึ่งวงกลม (semicircular canal system) ตรวจจับการเคลื่อนไหวแบบหมุน มีหลอดกึ่งวงกลม (semicircular canal) เป็นองค์ประกอบหลัก

โครงสร้าง

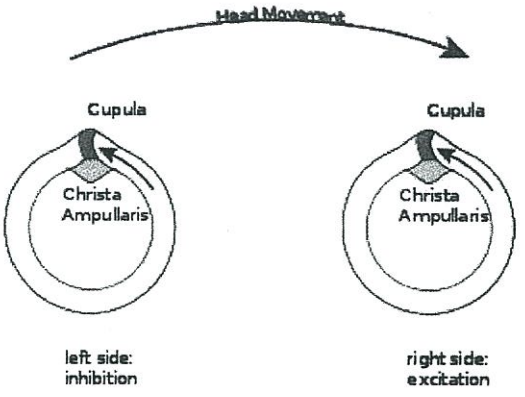
เนื่องจากว่าเราอยู่ในโลกมี 3 มิติ ดังนั้น vestibular system จึงมีหลอดกึ่งวงกลม 3 หลอดในห้องหูชั้นในแต่ละข้าง หลอดเหล่านี้ตั้งเป็นแนวฉากต่อกันและกัน และเรียกว่า หลอดกึ่งวงกลมแนวนอน (horizontal) หรือ หลอดกึ่งวงกลมด้านข้าง (lateral), หลอดกึ่งวงกลมด้านหน้า (anterior) หรือ หลอดกึ่งวงกลมด้านบน (superior), และหลอดกึ่งวงกลมด้านหลัง (posterior) หรือ หลอดกึ่งวงกลมด้านล่าง (inferior) ส่วนหลอดกึ่งวงกลมด้านหน้าและด้านหลังรวมกันอาจจะเรียกว่า หลอดกึ่งวงกลมแนวตั้ง (vertical) การเคลื่อนไหวของของเหลวภายในหลอดกึ่งวงกลมแนวนอน ทำให้สามารถตรวจจับการหมุนหัวรอบแกนแนวตั้ง เช่นในการหมุนตัว หลอดกึ่งวงกลมด้านหน้าและด้านหลังตรวจจับการหมุนหัวในระนาบแบ่งซ้ายขวา (sagittal) เช่นในการผงกหัว และในระนาบแบ่งหน้าหลัง (coronal) เช่นในการตีลังกาแบบล้อเกวียน ทั้งหลอดกึ่งวงกลมด้านหน้าและทั้งด้านหลังตั้งเป็นมุม 45° ระหว่างระนาบแบ่งหน้าหลังและระนาบแบ่งซ้ายขวา ระบบหลอดกึ่งวงกลมนั้นสามารถตรวจจับการเคลื่อนไหวได้ ก็เพราะการเคลื่อนไหวของของเหลวที่อยู่ในหลอดเข้าไปกดที่โครงสร้างชื่อว่า cupula ซึ่งมีเซลล์ขนที่ถ่ายโอนการเคลื่อนไหวเชิงกลไปเป็นสัญญาณไฟฟ้า

2.1.2 ทำงานเป็นระบบดันและดึง

หลอดกึ่งวงกลมแต่ละอันในศีรษะซีกซ้ายตั้งอยู่เกือบขนานกันกับหลอดในซีกขวา หลอดแต่ละคู่ทำงานคล้ายกับการใช้แรงดันแรงดึง คือ เมื่อหลอดหนึ่งส่งสัญญาณในระดับที่สูงขึ้น (เมื่อเกิดการเร้า) หลอดที่คู่กันอีกข้างหนึ่งก็จะส่งสัญญาณในระดับที่ต่ำลง (เมื่อเกิดการยับยั้ง) แม้ในนัยตรงกันข้ามกันก็เป็นอย่างนี้

ระบบดันและดึงนี้ ทำให้สามารถรู้สึกการหมุนศีรษะไปทั่วทุกทิศ คือ หลอดแนวนอนซีกขวารับการกระตุ้นในช่วงที่ศีรษะหมุนไปทางขวา (รูป 3) และหลอดแนวนอนซีกซ้ายก็รับการกระตุ้นในช่วงที่ศีรษะหมุนไปทางซ้าย ส่วนหลอดแนวตั้งจับคู่กันแบบไขว้ทแยง คือ เมื่อหลอดแนวตั้งด้านหน้ารับการกระตุ้น หลอดแนวตั้งด้านหลังของศีรษะอีกซีกหนึ่งก็จะรับการยับยั้ง แม้ในนัยตรงกันข้ามกันก็เป็นอย่างนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.2 ระบบดันและดึงของหลอดกึ่งวงกลมแสดงการทำงานเมื่อมีการหมุนหัวในแนวนอนไปทางขวา ซึ่งทำให้เกิดสัญญาณแบบเร้าที่หลอดกึ่งวงกลมแนวนอนด้านขวา และสัญญาณแนวยับยั้งที่หลอดกึ่งวงกลมแนวนอนด้านซ้าย

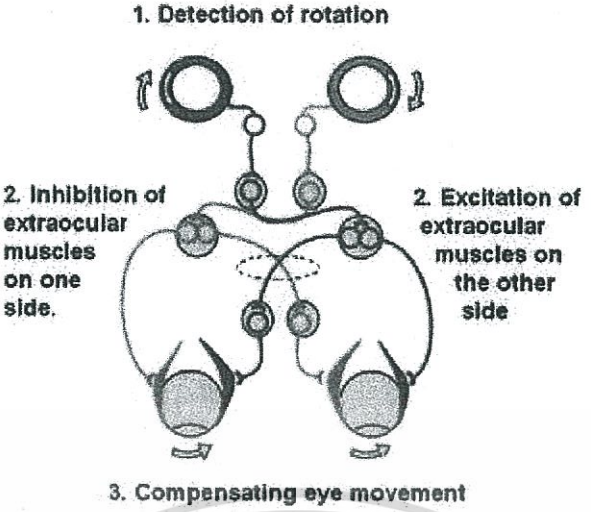
ที่มา: https://th.wikipedia.org/wiki/Vestibular_system

2.1.3 Vestibulo-Ocular Reflex (VOR)

vestibulo-ocular reflex (VOR) เป็นรีเฟล็กซ์การเคลื่อนไหวตาที่สร้างเสถียรภาพให้กับรูปที่ตกลงบนจอตาในขณะที่มีการเคลื่อนไหวศีรษะ โดยเคลื่อนตาไปในทิศทางตรงกันข้ามของการเคลื่อนศีรษะ และดังนั้นจึงรักษารูปไว้ให้ตกลงที่ตรงกลางของลานสายตา ยกตัวอย่างเช่น ถ้าศีรษะเคลื่อนไปทางขวา ตาทั้งสองก็จะหมุนไปทางซ้ายมีผลให้การทอดสายตาตกลงที่จุดเดิม (ดูรูป 4 หรือลองเพ่งที่ตัวอักษร "นี้" แล้วลองขยับหัว) และนัยตรงกันข้ามกันก็จะเป็นอย่างนี้ด้วย และเนื่องจากว่า มีการเคลื่อนไหวศีรษะอย่างเล็กน้อยอยู่อย่างตลอดเวลา VOR จึงสำคัญมากในการสร้างเสถียรภาพให้กับการเห็น คนไข้ที่มี VOR เสียหายจะอ่านหนังสือได้ยาก เพราะว่า ไม่สามารถรักษาการทอดสายตาลงที่จุดเดียวกันเมื่อศีรษะมีการขยับเล็ก ๆ น้อย ๆ VOR ทำงานได้โดยไม่ได้อาศัยการเห็น คือสามารถทำงานได้แม้ในที่ที่มืดสนิทหรือแม้เมื่อปิดตาอยู่

รีเฟล็กซ์นี้เมื่อใช้กับหลักการดันและดึงดังที่อธิบายมาแล้ว เป็นหลักทางกายภาพในวิธีการตรวจประสาทที่เรียกว่า Head Impulse Test (HIT) หรือที่เรียกว่า Halmagyi-Curthoys-test ที่มีการเคลื่อนศีรษะของผู้รับการทดสอบไปทางด้านข้างอย่างรวดเร็ว ในขณะที่ตรวจดูว่าตาทั้งสองของผู้รับการทดสอบมองไปที่ที่เดียวกันในเวลาเดียวกันหรือไม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



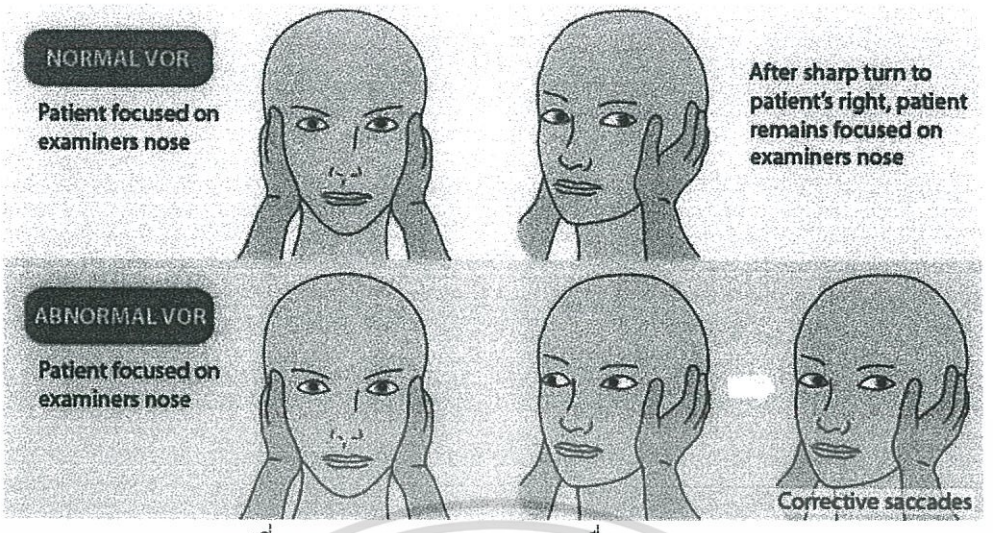
รูปที่ 2.3 แสดงการทำงานของ vestibulo-ocular reflex ตามลำดับตัวเลข (1) การหมุนของศีรษะไปทางขวา (2) ส่งสัญญาณยับยั้ง (inhibitory สีส้ม) ไปที่กล้ามเนื้อตาข้างหนึ่ง และส่งสัญญาณเร้า (excitatory สีน้ำเงิน) ไปยังกล้ามเนื้อตาอีกข้างหนึ่ง (3) ผลก็คือจะมีการเคลื่อนไหวตาไปทางซ้าย เพื่อชดเชยการเคลื่อนไหวของศีรษะ

ที่มา: https://th.wikipedia.org/wiki/Vestibular_system

2.1.4 Head Impulse Test (HIT) สำหรับ VOR

Head Impulse Test (HIT) [3] ถูกใช้ในการชี้วัดการทำงานของ VOR ว่ามีการทำงานปกติหรือขาดการทำงานของรีเฟล็กซ์นี้ วิธีนี้เป็นวิธีที่ทำได้ง่าย ทำได้โดยให้ผู้ป่วยนั่งตัวตรงและจ้องที่จุดๆ หนึ่ง (ส่วนมากคือปลายจมูกของผู้ทำการทดสอบหรือแพทย์) จากนั้นผู้ทดสอบจะทำการหันศีรษะผู้ป่วยอย่างรวดเร็วไปด้านซ้ายหรือขวา ผลลัพธ์เมื่อผู้ป่วยมีอาการผิดปกติคือเมื่อหัวศีรษะไปทางด้านใดด้านหนึ่งจะไม่สามารถกรอกตากลับมาที่จุดเดิม(ปลายจมูกของผู้ทำการทดสอบ) ได้โดยอัตโนมัติ ทำให้ผู้ป่วยต้องกรอตากลับมาด้วยตัวเอง (saccade) ซึ่งบ่งบอกว่าผู้ป่วยมีอาการของการเวียนศีรษะแบบบ้านหมุน (vertigo) ที่มีสาเหตุจากระบบ vestibular รูปที่ 2.4 แสดงการทำ HIT เพื่อทดสอบ VOR

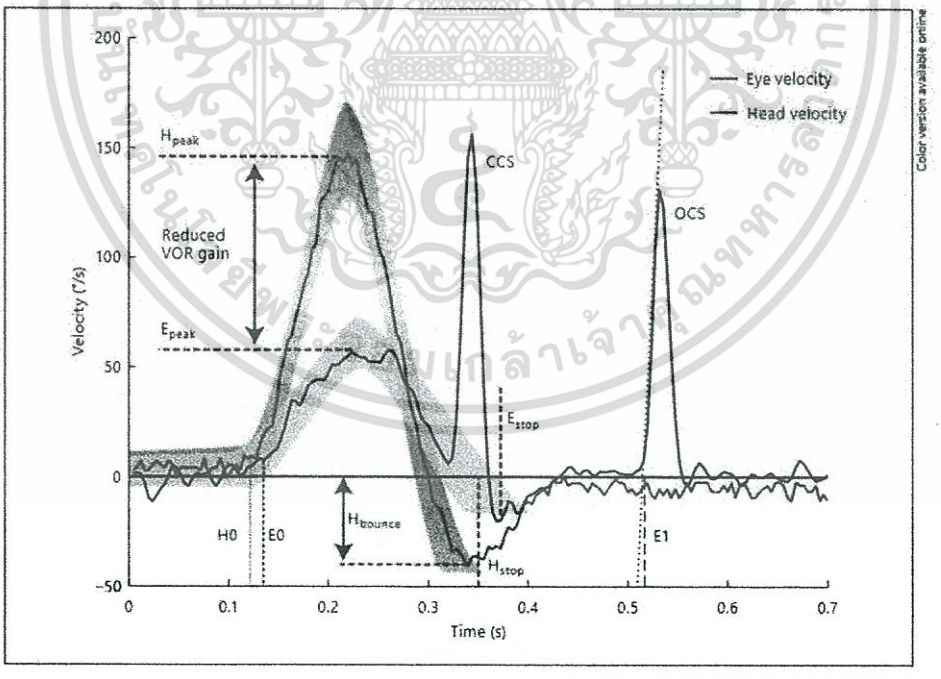
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.4 Head Impulse Test เพื่อทดสอบ VOR

ที่มา: https://bpac.org.nz/BPJ/2012/september/vertigo.aspx

จากงานวิจัยที่เกี่ยวข้อง [2] ความเร็วของการเคลื่อนที่ของศีรษะ (สีแดง) และดวงตา (สีดำ) แสดงดังรูปที่ 2.5 การกรอกตากลับมาของผู้ป่วย (overt corrective saccade, E1) หลังจากที่ศีรษะหยุด (H_{stop}) นั้น ใช้ระยะเวลาเท่ากับ E1-H_{stop} คือมีค่าประมาณ 100-200 ms



รูปที่ 2.5 กราฟความเร็วของศีรษะ (สีแดง) และดวงตา (สีดำ) ขณะทำ Head Impulse Test

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 การทรงตัว

การทรงตัว [4] หรือการรักษาสมดุลของการทรงตัว ทำให้คนเราสามารถนั่ง นอน ยืน เดิน วิ่ง ปฏิบัติกิจวัตรประจำวัน และปฏิบัติกิจกรรมนอกเหนือจากกิจวัตรประจำวัน เช่น การเล่นกีฬา ว่ายน้ำ ขับรถและกิจกรรมอื่น ๆ ที่เป็นเรื่องเฉพาะตัวได้อย่างปกตินั้น ต้องอาศัยกลไกของการทรงตัวหลายอย่างทำงานร่วมกัน ได้แก่ (1) การรับรู้สภาพแวดล้อมด้วยสายตา (Vision), (2) การรับรู้ตำแหน่งของร่างกายผ่านกล้ามเนื้อ ข้อของแขนขา และกระดูกสันหลัง (Kinesthetic Sense) และ (3) การรับรู้การเปลี่ยนแปลงตำแหน่งของศีรษะผ่านทางประสาททรงตัวในหูชั้นในทั้งสองข้าง (Vestibular End-Organ) โดยการทำงานของการรับรู้ทั้งสามนี้ จะต้องประสานกันอย่างสมดุล และส่งสัญญาณไปสู่ศูนย์รับและประมวลผลที่ระบบประสาทส่วนกลาง มีการติดต่อไปยังสมองส่วน Cerebrum เพื่อการรับรู้ความรู้สึกและการควบคุมการทรงตัวในสถานการณ์ต่าง ๆ ได้อย่างสมดุล โดยไม่เกิดอันตรายต่อร่างกาย ในส่วนของสมองส่วนท้าย (Cerebellum) จะมีการส่งข้อมูลมายังศูนย์กลางการทรงตัวในก้านสมองด้วย ทำให้คนเราสามารถทรงตัวในสภาพแวดล้อมได้อย่างเป็นปกติ การผิดปกติของการทรงตัวจะทำให้เกิดอาการ และอาการแสดงทางคลินิกที่แยกได้ว่าเป็นการผิดปกติของระบบปลายทางของประสาทหรือระบบกลไกปรับตัวของสมอง

ความผิดปกติของการทรงตัว อาจทำให้เกิดความรู้สึกว่าตนเองมีการเคลื่อนไหวผิดปกติ ทรงตัวไม่ได้ (Subjective) หรือรู้สึกว่าสิ่งแวดล้อมรอบตัวมีการเคลื่อนไหว (Objective) ความรู้สึกที่ผิดปกตินี้ อาจเป็นลักษณะหมุน, ดึงและดัน, หรือเอียง เมื่อมีการเคลื่อนไหวของศีรษะ

ความผิดปกติของการทรงตัวอาจจำแนกตามอาการได้ดังนี้

1. อาการรู้สึกหมุนหรือเวียนศีรษะ (Vertigo) เนื่องจากมีความผิดปกติของ Vestibular system และ Central nervous system components ซึ่งได้แก่ Cerebral cortex, Cerebellum, Brainstem, และการเคลื่อนไหวของลูกตา
2. ความรู้สึกงง ๆ หรือศีรษะเบาๆ (Giddiness or Lightheadedness) โดยไม่มีอาการรู้สึกหมุน
3. มีความผิดปกติของการยืนและเดินโดยมีและไม่มีอาการเวียนศีรษะร่วมด้วย

2.2.1 การสูญเสียการทรงตัวเนื่องจากความผิดปกติของ Vestibular system [5]

การสูญเสียสมรรถภาพของการทรงตัวเกิดขึ้นได้จากความผิดปกติในการทำงานของ Vestibular organ ที่อยู่ในหูชั้นใน (Labyrinthine) และการเชื่อมโยงกับระบบประสาทกลาง [สำหรับการสูญเสียการทรงตัวที่เกิดจากการทำงานผิดปกติของระบบประสาทกลาง (Cerebrum,

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Cerebellum, Brainstem) ให้ดูแนวทางการประเมินการสูญเสียสมรรถภาพในบทที่ 5 ระบบประสาท)] การทำงานผิดปกติของ Vestibular organ อาจเกิดขึ้นได้ 2 ลักษณะ คือ

2.2.1.1. การสูญเสียหน้าที่โดยสิ้นเชิง (Complete Loss of Vestibular Function)

การสูญเสียหน้าที่โดยสิ้นเชิง (Complete Loss of Vestibular Function) อาจเป็นข้างเดียวหรือทั้งสองข้าง ถ้าเป็นข้างเดียว อาจมีการชดเชยการทรงตัวที่มากพอด้วยการทำงานระบบประสาทกลางได้ แต่ในกรณีที่เป็นทั้งสองข้าง การทรงตัวต้องอาศัยการรับรู้ตำแหน่งของร่างกาย (Kinesthetic or Proprioceptive Senses) และการมองเห็น (Vision) เท่านั้น ซึ่งโดยทั่ว ๆ ไป จะไม่สามารถชดเชยได้สมบูรณ์จนทำให้บุคคลมีการเคลื่อนไหวหรือเคลื่อนที่ได้เป็นปกติได้ การสูญเสียหน้าที่โดยสิ้นเชิงของ Vestibular system จะทำให้มีการสูญเสียสมรรถภาพตั้งแต่ร้อยละ 0-95 ของทั้งร่างกาย ทั้งนี้ขึ้นอยู่กับความสามารถในการประกอบกิจวัตรประจำวันของบุคคลที่มีการสูญเสียดังกล่าว

2.2.1.2. การแปรปรวนในการทำหน้าที่ (Disturbances of Vestibular Function)

การแปรปรวนในการทำหน้าที่จะทำให้เกิดอาการรู้สึกหมุนหรือเวียนศีรษะ (Vertigo or Vestibular Dysequilibrium) การรู้สึกงงหรือศีรษะเบา (Giddiness or Lightheadedness) และการเดินผิดปกติ โดยไม่มีอาการเวียนศีรษะร่วมด้วย ไม่จัดเป็นการแปรปรวนในการทำหน้าที่ของ Vestibular organ อาการเวียนศีรษะ อาจมีอาการอื่นร่วมด้วยซึ่งได้แก่อาการคลื่นไส้ อาเจียน ปวดศีรษะ เคลื่อนไหวไม่ได้เดินเซและมีการกระตุกของลูกตา (Nystagmus) การเคลื่อนไหวร่างกาย อาจทำให้อาการเวียนศีรษะและอาการร่วมอื่นๆ รุนแรงขึ้นการแปรปรวนในการทำหน้าที่ของ Vestibular organ (Labyrinthine) มักเกิดขึ้นร่วมกับการสูญเสียการได้ยิน (Hearing Loss) และเสียงดังรบกวนในหู(Tinnitus) เนื่องจากการแปรปรวนในการทำหน้าที่อาจเกิดขึ้นชั่วคราวหรือถาวร ดังนั้น การประเมินการสูญเสียสมรรถภาพจึงควรทำเมื่อผู้รับการประเมินมีอาการคงที่ หรือมีการปรับตัวอย่างเต็มที่แล้ว ซึ่งมักต้องใช้เวลาหลายเดือน หลังจากได้รับบาดเจ็บหรือเป็นโรค

2.2.2. การประเมินการสูญเสียสมรรถภาพของการทรงตัว

การประเมินผู้ป่วยที่มีการสูญเสียสมรรถภาพของการทรงตัว แบ่งการสูญเสียเป็น 5 ระดับ แต่ละระดับจะเป็นเกณฑ์ที่แสดงสมรรถภาพที่เหลือของผู้ป่วยต่อการใช้ชีวิตประจำวันตามปกติ การประเมินความรุนแรงของการสูญเสียสมรรถภาพตามการจัดระดับนี้ อยู่บนพื้นฐานว่า ผู้รับการประเมินได้รับการวินิจฉัย โรคที่ถูกต้อง จากการซักประวัติอย่างรอบคอบ และการตรวจทางคลินิกที่สนับสนุนอย่างครบถ้วน โดยไม่อาศัยคำบอกเล่าของผู้รับการประเมินเพียงอย่างเดียว เสริมด้วยการใช้วิจารณญาณอย่างมีเหตุผลของแพทย์ในทางคลินิก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระดับที่ 1 สูญเสียสมรรถภาพร้อยละ 0 ของทั้งร่างกาย

1. มีอาการหรืออาการแสดงของการเสียการทรงตัว แต่ไม่มีหลักฐานสนับสนุนจากการตรวจที่เป็นภววิสัย (Objective Finding) และ

2. สามารถประกอบกิจวัตรประจำวันได้ตามปกติโดยไม่ต้องมีการช่วยเหลือ

ระดับที่ 2 สูญเสียสมรรถภาพร้อยละ 1-10 ของทั้งร่างกาย

1. มีอาการหรืออาการแสดงของการเสียการทรงตัว โดยมีหลักฐานสนับสนุนจากการตรวจที่เป็นภววิสัย (Objective Finding) และ

2. สามารถประกอบกิจวัตรประจำวันได้ตามปกติโดยไม่ต้องมีการช่วยเหลือ นอกจากกิจกรรมซับซ้อนบางอย่าง เช่น การขี่จักรยาน หรือการทำงานบางอย่างของบุคคลนั้น เช่น การเดินบนคาน หรือ นั่งร้านในการก่อสร้าง เป็นต้น

ระดับที่ 3 สูญเสียสมรรถภาพร้อยละ 11-30 ของทั้งร่างกาย

1. มีอาการหรืออาการแสดงของการเสียการทรงตัว โดยมีหลักฐานสนับสนุนจากการตรวจที่เป็นภววิสัย (Objective Finding) และ

2. ไม่สามารถประกอบกิจวัตรประจำวันได้ตามปกติ ถ้าปราศจากการช่วยเหลือ ยกเว้นกิจกรรมง่าย ๆ เช่น การดูแลตนเอง (Self-care), งานบ้านบางอย่าง, การเดิน, การนั่งยานยนต์ที่คนอื่นขับ

ระดับที่ 4 สูญเสียสมรรถภาพร้อยละ 31-60 ของทั้งร่างกาย

1. มีอาการหรืออาการแสดงของการเสียการทรงตัว โดยมีหลักฐานสนับสนุนจากการตรวจที่เป็นภววิสัย (Objective Finding) และ

2. บุคคลนั้นไม่สามารถประกอบกิจกรรมในชีวิตประจำวันได้ตามปกติลำพัง โดยไม่มีการช่วยเหลือจากบุคคลอื่น นอกจาก การดูแลตนเอง

ระดับที่ 5 สูญเสียสมรรถภาพร้อยละ 61-95 ของทั้งร่างกาย

1. มีอาการหรืออาการแสดงของการเสียการทรงตัว โดยมีหลักฐานสนับสนุน จากการตรวจที่เป็นภววิสัย (objective finding) และ

2. ไม่สามารถประกอบกิจวัตรในชีวิตประจำวันได้ตามลำพังโดยไม่มีการช่วยเหลือจากบุคคลอื่น นอกจากการดูแลตนเองที่ไม่ต้องเคลื่อนย้ายตัวเอง (Ambulation) และ

3. จำเป็นต้องอยู่แต่ในบ้านหรือสถานที่พักอื่น ๆ เท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 อาการเวียนศีรษะ

อาการเวียนศีรษะ [6,7] เป็นคำที่มีความหมายค่อนข้างกว้าง อาจหมายรวมถึงอาการมึนศีรษะ วิงเวียน งง รู้สึกโคลงเคลง ทรงตัวไม่ค่อยได้ มีความรู้สึกลอยๆ หวิวๆ มีอาการตื้อในศีรษะ ซึ่งในทางการแพทย์จะแบ่งอาการนี้ออกเป็น 2 ลักษณะ คือ

- อาการมึนเวียนศีรษะ (dizziness) มีความหมายรวมตั้งแต่อาการมึนศีรษะไปจนถึงอาการวิงเวียนศีรษะ ซึ่งเป็นอาการไม่เฉพาะเจาะจง เกิดได้จากโรคต่างๆ เช่น โรคทางระบบไหลเวียนเลือด โรคทางระบบประสาท ภาวะโลหิตจาง เป็นต้น
- อาการเวียนศีรษะบ้านหมุน (vertigo) จะหมายถึงเฉพาะอาการเวียนศีรษะแบบรู้สึกหมุนหรือโคลงเคลงเท่านั้น

อาการเวียนศีรษะบ้านหมุน (vertigo) เป็นอาการที่มีสาเหตุมาจากความผิดปกติของอวัยวะการทรงตัวในหูชั้นใน ซึ่งเป็นส่วนที่ทำหน้าที่คอยรับการทรงตัวสมดุลของร่างกายในท่าทางต่างๆ เมื่อเกิดความผิดปกติขึ้นจึงทำให้มีอาการเวียนศีรษะแบบรู้สึกหมุน ผู้ป่วยจะรู้สึกว่าสิ่งแวดล้อมหมุนรอบตัวเองหรือตัวเองหมุน รู้สึกโคลงเคลงทั้งๆที่ตัวเองอยู่กับที่หรือไม่มีการเคลื่อนไหว ในรายที่มีอาการรุนแรงมากอาจมีอาการคลื่นไส้ อาเจียน มีความรู้สึกเหมือนจะเป็นลม หูอื้อ การได้ยินลดลง หรือมีเสียงในหูร่วมด้วยได้

ทั้งนี้ มีหลายโรคที่เป็นสาเหตุให้เกิดอาการเวียนศีรษะบ้านหมุนได้ อาทิ

- โรคหินปูนในหูชั้นในเคลื่อน หรือโรคเวียนศีรษะขณะเปลี่ยนท่า (benign paroxysmal positioning vertigo: BPPV) เป็นโรคที่ทำให้เกิดอาการเวียนศีรษะบ้านหมุนที่พบได้บ่อยที่สุด โรคนี้เป็นโรคที่เกิดจากความเสื่อมของหูชั้นใน จึงพบมากในผู้สูงอายุ อาการเฉพาะของโรคนี้คือ อาการเวียนศีรษะบ้านหมุนที่เกิดขึ้นทันทีทันใดในขณะเปลี่ยนท่าทางของศีรษะ เช่น ระหว่างกำลังล้มตัวลงนอนหรือลุกจากที่นอน เงยหน้า ก้มหยิบของ เป็นต้น อาการมักจะเป็นระยะเวลาสั้นๆ เป็นแค่ช่วงวินาทีที่ขยับศีรษะ แล้วอาการจะค่อยๆ หายไป ผู้ป่วยโรคนี้จะไม่มีอาการหูอื้อ ไม่พบการสูญเสียการได้ยินหรือเสียงผิดปกติในหู (ยกเว้นในรายที่เป็นโรคหูอยู่ก่อนแล้ว) รวมถึงไม่มีอาการทางระบบประสาท เช่น แขนขาชาหรืออ่อนแรง
- โรคน้ำในหูชั้นในผิดปกติหรือโรคน้ำในหูไม่เท่ากัน (Meniere's disease) เป็นโรคที่เกิดจากความผิดปกติของหูชั้นใน โดยยังไม่ทราบสาเหตุที่แน่ชัด แต่พบว่าอาการของโรคเป็นผลจากความผิดปกติของน้ำที่อยู่ภายในหูชั้นใน ซึ่งทำให้ผู้ป่วยเกิดอาการเวียนศีรษะแบบรู้สึกหมุนอย่างรุนแรง ร่วมกับมีอาการคลื่นไส้อาเจียนและสูญเสียสมดุลของร่างกาย ทำให้เซหรือล้มได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ง่าย อาการเวียนศีรษะที่เกิดจากโรคนี้อาจนานเป็นนาทีจนถึงหลายชั่วโมง ซึ่งในระหว่างที่เกิดอาการ ผู้ป่วยควรอยู่นิ่งๆ ไม่ขยับศีรษะ เพราะอาจทำให้มีอาการเวียนศีรษะเพิ่มขึ้นได้ นอกจากนี้ผู้ป่วยยังอาจมีการได้ยินลดลงและมีเสียงดังในหู บางครั้งอาจพบอาการหูอื้อได้ด้วย

- การอักเสบของหูชั้นใน (labyrinthitis) พบการอักเสบจากเชื้อไวรัสได้บ่อย ซึ่งมักมีประวัติการเป็นหวัดหรือระบบทางเดินหายใจอักเสบนำมาก่อน ถ้าเชื้อไวรัสลามเข้าสู่หูชั้นในและเส้นประสาท จะทำให้เกิดการอักเสบ ซึ่งทำให้ผู้ป่วยมีอาการเวียนศีรษะรุนแรงและเป็นอยู่หลายวัน แต่ผู้ป่วยมักมีการได้ยินที่ปกติ แต่หากเป็นการอักเสบที่เกิดจากเชื้อแบคทีเรีย ซึ่งมักพบในผู้ป่วยที่มีประวัติโรคการอักเสบของหูชั้นกลาง โรคหูน้ำหนวก แล้วลุกลามเข้าสู่หูชั้นใน อาการมักรุนแรงมาก ผู้ป่วยมักมีอาการสูญเสียการได้ยินร่วมด้วย
- โรคเนื้องอกของประสาทการทรงตัวหรือเส้นประสาทการได้ยิน (acoustic neuroma) ผู้ป่วยจะมีอาการเวียนศีรษะร่วมกับการได้ยินลดลง บางรายอาจมีเสียงรบกวนในหู สำหรับรายที่มีเนื้องอกขนาดใหญ่และไม่ได้รับการรักษา ผู้ป่วยอาจมีอาการชาที่ใบหน้าซีกนั้น อัมพาตของใบหน้า เดินโซเซ หรืออาการทางสมองอื่นๆ เนื่องจากก้อนเนื้องอกไปกดทับเนื้อสมอง
- โรคเส้นประสาทการทรงตัวในหูอักเสบ (vestibular neuronitis) ทำให้เกิดอาการเวียนศีรษะรุนแรงนานหลายวันจนถึงสัปดาห์ แต่ไม่ส่งผลต่อการได้ยิน ผู้ป่วยยังคงได้ยินเป็นปกติ
- กระดูกกะโหลกแตกหัก (temporal bone fracture)
- เลือดไปเลี้ยงสมองไม่เพียงพอ (vertebra-basilar insufficiency)

อาการเวียนศีรษะบ้านหมุนมีสาเหตุที่ซับซ้อน การวินิจฉัยโรคอย่างถูกต้องจึงมีความสำคัญการรักษาในระยะแรกจะได้ผลดีกว่าในระยะหลัง ดังนั้นหากมีอาการเวียนศีรษะผิดปกติ ควรรีบพบแพทย์เพื่อหาสาเหตุของอาการและเข้ารับการรักษา โดยแพทย์จะทำการวินิจฉัยด้วยการซักประวัติและตรวจร่างกายอย่างละเอียด ทั้งการตรวจหู ตรวจระบบประสาทและการทรงตัว ตรวจการทำงานของอวัยวะการทรงตัวในหูชั้นใน ตรวจดูการกลอกของลูกตาและการเคลื่อนไหวของลูกตาในท่าทางต่างๆ ในผู้ป่วยบางรายที่แพทย์สงสัยว่ามีความผิดปกติของการทำงานในหูชั้นในอาจได้รับการตรวจพิเศษเพิ่มเติม เช่น

- ตรวจการได้ยิน (audiogram)
- ตรวจการทำงานของอวัยวะทรงตัวของหูชั้นใน (video electronystagmography: VNG)
- ตรวจวัดแรงดันของน้ำในหูชั้นใน (electrocochleography: ECOG)
- ตรวจการทรงตัว (post urography)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ตรวจการทำงานของเส้นประสาทการได้ยิน (evoke response audiometry) เป็นต้น

สำหรับการรักษา แพทย์จะพิจารณารักษาตามสาเหตุที่ก่อให้เกิดอาการเวียนศีรษะบ้านหมุน ซึ่งแนวทางการรักษาจะแตกต่างกันไป โดยแพทย์จะพิจารณาแนวทางการรักษาที่เหมาะสมสำหรับผู้ป่วยแต่ละราย

การดูแลตนเองและพยายามหลีกเลี่ยงปัจจัยกระตุ้นก็ยังคงมีความสำคัญต่อการรักษาและการป้องกันการเกิดโรค ในผู้ที่มีอาการเวียนศีรษะบ้านหมุนควรปฏิบัติตัวดังนี้

- หลีกเลี่ยงท่าทางที่ทำให้เกิดอาการเวียนศีรษะในระหว่างเกิดอาการ เช่น การหมุนหันศีรษะเร็วๆ การเปลี่ยนแปลงท่าทางอิริยาบถอย่างรวดเร็ว การก้มเงยคอหรือหันอย่างเต็มที่
- ลดปริมาณหรืองดการสูบบุหรี่/ดื่มกาแฟ
- หลีกเลี่ยงปัจจัยกระตุ้นที่ทำให้เกิดอาการเวียนศีรษะ เช่น ความเครียด ความวิตกกังวล สารก่อภูมิแพ้ต่างๆ และการพักผ่อนไม่เพียงพอ
- ไม่ควรอยู่ในสถานการณ์ที่เสี่ยงต่อการเกิดอุบัติเหตุ เช่น การขับขี่ยานพาหนะในขณะยังมีอาการ การปีนป่ายที่สูง

2.4 Python Programming Language

ภาษาไพทอน [8,9] (Python programming language) เป็นภาษาระดับต่ำ และเป็น Open Source ทำให้ทุกคนสามารถที่จะนำ Python มาพัฒนาโปรแกรมได้โดยไม่ต้องเสียค่าใช้จ่าย และความเป็น Open Source ทำให้มีคนเข้ามาช่วยกันพัฒนาให้ Python มีความสามารถสูงขึ้น และใช้งานได้ครบคุมกับทุกลักษณะงานPython ถูกสร้างขึ้นมาโดย Guido Van Rossum โค้ดของ Python ถูกสร้างขึ้นมาจากภาษาซี การประมวลผลจะทำในแบบอินเทอร์พรีเตอร์ คือจะประมวลผลไปทีละบรรทัดและปฏิบัติตามคำสั่งที่ได้รับ Python เวอร์ชันแรกคือ 0.9.0 ออกมาเมื่อปี 2533 และเวอร์ชันปัจจุบันคือ 3.6.3 (24 ตุลาคม 2560)

คุณลักษณะเด่นของภาษา Python

- ความเป็นภาษาสคริปต์

เนื่องจากไพทอนเป็นภาษาสคริปต์ ทำให้ใช้เวลาในการเขียนและคอมไพล์ไม่มาก ทำให้เหมาะกับงานด้านการดูแลระบบ (System administration) เป็นอย่างยิ่ง ได้มีการสนับสนุนภาษาไพทอนโดยเป็นส่วนหนึ่งของระบบปฏิบัติการยูนิกซ์, ลินุกซ์ และสามารถติดตั้งให้ทำงานเป็นภาษาสคริปต์ของ Windows ผ่านระบบ Windows Script Host ได้อีกด้วย

- ไวยากรณ์ที่อ่านง่าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไวยากรณ์ของไพทอนได้กำจัดการใช้สัญลักษณ์ที่ใช้ในการแบ่งบล็อกของโปรแกรม และใช้การย่อหน้าแทน ทำให้สามารถอ่านโปรแกรมที่เขียนได้ง่าย นอกจากนั้นยังมีการสนับสนุนการเขียน docstring ซึ่งเป็นข้อความสั้นๆที่ใช้อธิบายการทำงานของฟังก์ชัน, คลาส, และโมดูลอีกด้วย

- ความเป็นภาษากาว

ไพทอนเป็นภาษากาว (Glue Language) ได้อย่างดีเนื่องจากสามารถเรียกใช้ภาษาโปรแกรมอื่นๆ ได้หลายภาษา ทำให้เหมาะที่จะใช้เขียนเพื่อประสานงานโปรแกรมที่เขียนในภาษาต่างกันได้

การนำไปใช้งาน

- ภาษา Python สนับสนุนแนวแบบคิด OOP (Object Oriented Programming)
- Python เป็นภาษาคอมพิวเตอร์ที่ไม่คิดมูลค่าการใช้งานและเป็นภาษาที่มีความยืดหยุ่นสูงมาก
- โค้ดที่เขียนด้วย Python สามารถนำไปรันบนระบบปฏิบัติการอื่นๆ ได้(Portable)เช่น Linux, MS-windows (95, 98, NT, 2000, XP), Amiga, Be-OS, OS/2, VMS, QNX, และระบบอื่นๆอีกมากมาย
- Python สนับสนุนเทคโนโลยี COM ของ MS-windows
- Python รวมมาตรฐานการอินเตอร์เฟส Tkinter ซึ่งสนับสนุนบนระบบ X windows, MS-windows และ Macintosh การใช้คำสั่ง Tkinter API ช่วยให้โปรแกรมเมอร์ไม่ต้องแก้ไขโค้ดเมื่อนำไปรันบนระบบปฏิบัติการอื่นๆ
- Python เป็น Dynamic typing คือ สามารถเปลี่ยนชนิดข้อมูลได้ง่ายและสะดวก
- Python มี Build-in Object Types คือ โครงสร้างของข้อมูลที่สามารถใช้ได้ใน Python ประกอบด้วย ลิสต์, ดิกชันนารี, สตริง ที่ง่ายต่อการใช้งานและมีประสิทธิภาพสูง
- Pythonมีเครื่องมือต่างๆ มากมาย เช่น การประมวลผลเท็กซ์ไฟล์ การเรียงข้อมูล การเชื่อต่อสตริง การตรวจสอบเงื่อนไขของข้อความ การแทนคำ เป็นต้น
- Python มีมอดูลสำหรับจัดการ Regular Expression
- Python มีมอดูลที่สร้างขึ้นจากนักพัฒนาสนับสนุนมากมาย ได้แก่ COM, Image, CORBA, ORBs, XML เป็นต้น
- Python จัดการหน่วยความจำอย่างอัตโนมัติ สามารถจักการพื้นที่หน่วยความจำที่ไม่ต่อเนื่องให้ทำงานได้อย่างมีประสิทธิภาพ
- Python อนุญาตให้ฝังชุดคำสั่งของ Python เอาไว้ภายในโคดภาษา C/C++ ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Python อนุญาตให้โปรแกรมเมอร์สร้าง Dynamic Link Library (DLL) เพื่อใช้ร่วมกับ Python
- Python มีมอดูลสนับสนุนเกี่ยวกับ Network process thread regular, expression, xml, GUI และอื่นๆ
- Pythonประกอบด้วยมอดูลสำหรับสร้าง Internet Script และติดต่อกับอินเทอร์เน็ตผ่าน Sockets, และทำหน้าที่เป็น CGI Script ตลอดจนใช้งานคำสั่ง FTP, Glopher, XML และอื่นๆอีกมาก
- Python สามารถประมาลผมทางด้านวิยาศาสตร์ และวิศวกรรมศาสตร์ได้อย่างมีประสิทธิภาพ
- Python มีฟังก์ชันสนับสนุนฐานข้อมูล เช่น MySQL, Sybase, Oracle, Informix, ODBC และอื่นๆ
- Python มีไลบรารีสนับสนุนด้านการสร้างภาพกราฟฟิก เช่น ทำภาพเบลอ หรือภาพชัด หรือเขียนข้อความบนภาพ ตลอดจนบันถึกไฟล์ในรูปแบบต่างๆ ได้อย่างสะดวกและมีประสิทธิภาพ
- Python มีไลบรารีสนับสนุนด้านปัญญาประดิษฐ์
- Python มีไลบรารีสำหรับสร้างเอกสาร PDF โดยไม่ต้องติดตั้ง Acrobat Writer
- Python มีไลบรารีสำหรับสร้าง Shockwaves Flash (SWF)

ด้วยความยืดหยุ่นของภาษา Python และความเป็นภาษาสคริปต์ทำให้มีการใช้งานไพทอนอย่างกว้างขวาง อย่างเช่นในองค์กรสำคัญต่างๆ

- Google ใช้ภาษาไพทอนในบริการหลายอย่าง เช่น Gmail, Google Map
- BitTorrent
- Chandler โปรแกรมจัดการข้อมูลส่วนบุคคล
- บางส่วนของ GNOME
- บางส่วนของ Blender
- Mailman โปรแกรมจัดการจดหมายกลุ่ม (เมลลิ่งลิสต์)
- MoinMoin โปรแกรมวิกิ
- Portage ส่วนจัดการแพกเกจของ Gentoo Linux
- Zope แอปพลิเคชันเซิร์ฟเวอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- เทอร์โบเกียร์ กรอบงานขนาดใหญ่สำหรับพัฒนาโปรแกรมประยุกต์บนเว็บ
- Django กรอบงานขนาดใหญ่สำหรับพัฒนาโปรแกรมประยุกต์บนเว็บ
- Red Hat ใช้ Python เป็นเครื่องมือสำหรับการติดตั้งระบบปฏิบัติการ Linux (Installer)
- Google สร้างขึ้นด้วย Python
- Infoseek ใช้ Python จัดการ web search products
- Yahoo! ใช้ Python ในการจักการด้าน Internet services
- NASA ใช้ Python สำหรับ mission-control-system
- Lawrence Livermore Labs ใช้ Python สำหรับงาน numeric programming
- Industrial Light and Magic ใช้ Python สร้างภาพแอนิเมชัน

2.5 OpenCV

OpenCV [10,11] ย่อมาจาก Intel® Open Source Computer Vision Library ประกอบด้วยฟังก์ชัน ภาษา C++ มากกว่า 300 ฟังก์ชันที่ใช้ในการประมวลภาพดิจิตอลที่สำคัญ OpenCV จัดเป็นโปรแกรมประเภท Freeware

OpenCV เป็น library โดยมีเป้าหมายหลักคือการใช้ประมวนผลภาพด้วยคอมพิวเตอร์แบบ real-time (real-time computer vision) โดยเริ่มนั้นถูกพัฒนาโดย Intel ต่อมาถูกพัฒนาโดย Willow Garage และปัจจุบันคือ Itseez นอกจากนั้น library ยังสามารถทำงานแบบ cross-platform นอกจากนั้นยังสนับสนุนการทำ Deep Learning frameworks เช่น TensorFlow, Torch/PyTorch และ Caffe.

OpenCV ถูกเขียนจากภาษา C++ และยังสามารถใช้ได้กับภาษา Python, Java และ MATLAB/OCTAVE และยังมีการพัฒนาเพื่อใช้สำหรับภาษา C#, Perl, Ch, Haskell และ Ruby

OpenCV สามารถใช้ได้กับคอมพิวเตอร์หลายระบบปฏิบัติการ: Windows, Linux, macOS, FreeBSD, NetBSD, OpenBSD รวมถึงระบบปฏิบัติการบนโทรศัพท์มือถือ: Android, iOS, Maemo, BlackBerry 10. ผู้ใช้สามารถโหลด OpenCV ได้ทั่วไปจาก SourceForge หรือ GitHub

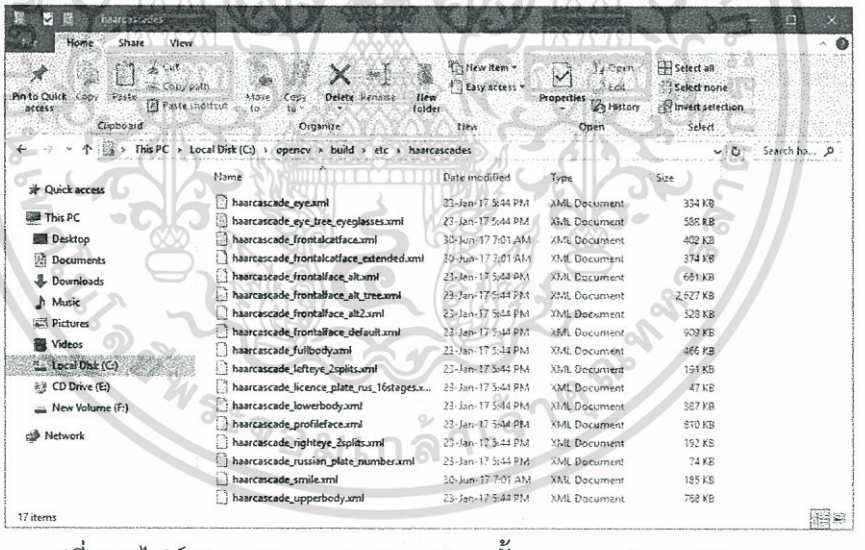
2.5.1 Haar Cascade Classifier

Haar Cascade Classifier (เรียกย่อๆว่า Classifier) หมายถึงหน่วยที่จะบอกว่าส่วนไหนของภาพเป็นสิ่งที่เราสนใจ Classifier จะถูกเทรนให้รู้จักวัตถุด้วยรูปของวัตถุจำนวนมาก (เช่น ใบหน้า รถยนต์ โลโก เป็นต้น) ซึ่งเรียกว่า Positive Samples ซึ่งถูกสเกลให้มีขนาดเท่ากัน เช่น 20 x 20 เป็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ต้น นอกจากนั้น Classifier ยังถูกเทรนให้รู้จักรูปที่ไม่ใช่วัตถุซึ่งหมายถึงภาพพื้นหลังที่มักพบวัตถุที่สนใจ เราเรียกว่า Negative Samples คำว่า Cascade Classifier หมายความว่า Classifier ที่ใช้ประกอบจาก Classifier แบบง่ายๆ (Simpler Classifier) ต่อกันเป็นลำดับชั้น การใช้ Classifier จะเริ่มจาก Classifier ลำดับแรกจนถึงลำดับชั้นหลังจนกระทั่งวัตถุไม่ผ่านในบางชั้นซึ่งไม่ถือว่าพบวัตถุ หรือผ่านทุกชั้นซึ่งถือว่าพบวัตถุ หลังจาก Classifier ถูกเทรนแล้ว เราสามารถใช้ Classifier ในการตรวจสอบว่าในภาพมีวัตถุที่เราสนใจหรือไม่ โดย Classifier จะถูกสแกนทั่วทั้งภาพโดยใช้หน้าต่าง เนื่องจากวัตถุที่สนใจอาจมีขนาดต่างกัน Classifier ถูกออกแบบให้สามารถสเกลให้มีขนาดต่างๆ เพื่อเพิ่มประสิทธิภาพในการค้นหา โดยทั่วไปการสแกนถูกทำหลายครั้ง แต่ละครั้งในสเกลที่ต่างๆ

OpenCV มีไฟล์ Haar Cascade Classifier สำหรับส่วนใบหน้า ส่วนร่างกาย ไฟล์ทั้งหมดแสดงในรูปที่ 2.6 ซึ่งได้มาจากการเทรนใบหน้าจากฐานข้อมูลขนาดใหญ่ รวมทั้งภาพพื้นหลังจำนวนมาก ซึ่งสามารถนำมาใช้งานในการประยุกต์ใช้งานการตรวจจับใบหน้าหรือร่างกาย ตัวอย่างผลลัพธ์การตรวจจับใบหน้าแสดงดังรูปที่ 2.7



รูปที่ 2.6 ไฟล์ Haar Cascade Classifier ทั้งหมดจาก OpenCV

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2.7 ตัวอย่างการตรวจจับใบหน้าด้วย Haar Cascade Classifer

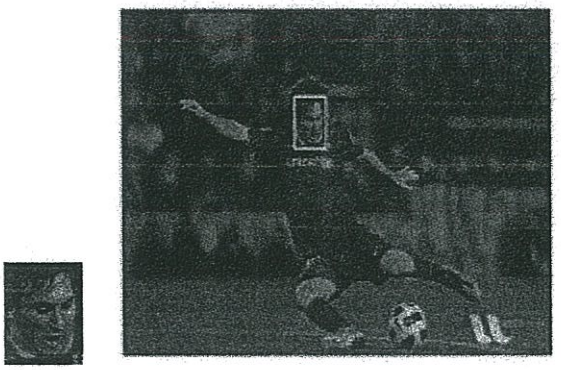
2.5.2 Template matching

Template Matching [12,13] เป็นวิธีการค้นหาตำแหน่งของภาพแม่แบบ (template image; T) ในภาพที่ใหญ่กว่า (source image; I) ซึ่งสำหรับ OpenCV ใช้ฟังค์ชัน cv.matchTemplate() การทำงานของฟังค์ชันคือการ slide ภาพแม่แบบไปบนภาพใหญ่ (2D convolution) และหาความแตกต่างแต่ละ pixel กับภาพแม่แบบ ในแต่ละตำแหน่ง (x,y) จะมีการคำนวณค่าออกมาเป็นเมทริกซ์ผลลัพธ์ R ซึ่งจะบอกค่าความเหมือนหรือต่างของตำแหน่ง (x,y) บนภาพกับภาพแม่แบบ โดย

$$R(x,y) = \frac{\sum_{x\prime,y\prime}(T(x',y') \cdot I(x+x',y+y'))}{\sqrt{\sum_{x\prime,y\prime} T(x',y')^2 \cdot \sum_{x\prime,y\prime} I(x+x',y+y')^2}}$$

ซึ่งหากภาพ (I) มีขนาด WxH และ ภาพแม่แบบ (T) มีขนาด w*h ผลลัพธ์ที่ออกากจะได้ภาพขนาด W-w+1xH-h+1 เมื่อได้ผลลัพธ์ R ออกมาเราสามารถใช้ฟังค์ชัน cv.minMaxLoc() กับเมทริกซ์ R เพื่อหาค่าสูงสุด/ต่ำสุดและใช้จุดที่ค่าสูงสุดเป็นจุดขวาบนเพื่อวาดสี่เหลี่ยมขนาด wxh ซึ่งเป็นตำแหน่งของภาพแม่แบบนั่นเอง รูปที่ 2.8 แสดงตัวอย่างผลลัพธ์การทำงานของ template matching

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.8 การทำ template matching (ซ้าย) แม่แบบ (template) (ขวา) ผลลัพธ์การตรวจจับ

ที่มา: https://docs.opencv.org/trunk/d4/dc6/tutorial_py_template_matching.html

2.6 Dlib library

Dlib [14,15] เป็น library ที่สามารถทำงานแบบ cross-platform ที่เขียนขึ้นมาจากภาษา C++ และเป็น open-source โดย Dlib สามารถนำไปใช้ได้หลากหลายด้าน ด้วยการออกแบบโดย component-based software engineering

Davis King เป็นผู้ริเริ่มสร้าง dlib ตั้งแต่ปีค.ศ. 2002 ซึ่งแรกเริ่มนั้นได้พัฒนาในหลากหลายด้าน เช่น เครือข่าย, กราฟฟิก, โครงข่ายข้อมูล, พีชคณิต, machine learning, การประมวนผลภาพ และอื่นๆ ซึ่งต่อมาถูกพัฒนาโดยเน้นไปที่ machine learning แต่ยังคงสามารถนำไปใช้ประโยชน์ได้หลากหลายด้าน

ส่วนสำคัญของปรัชญาในการพัฒนา Dlib คือการพัฒนาเพื่ออากรใช้งานที่ง่ายและสะดวก ดังนั้น code ทั้งหมดของ Dlib จะถูกออกแบบให้ใช้ง่ายที่สุดเท่าที่จะทำได้และไม่ต้องการการติดตั้งหรือปรับตั้งค่าต่างๆ การที่จะทำเช่นนั้น ทุก ๆ code จะมีการทำ API (Application Programming Interface) wrapper หรือเขียนด้วยภาษา C++ ปัจจุบัน Dlib สามารถทำงานได้บน OS X, MS Windows, Linux, Solaris, BSDs และ HP-UX

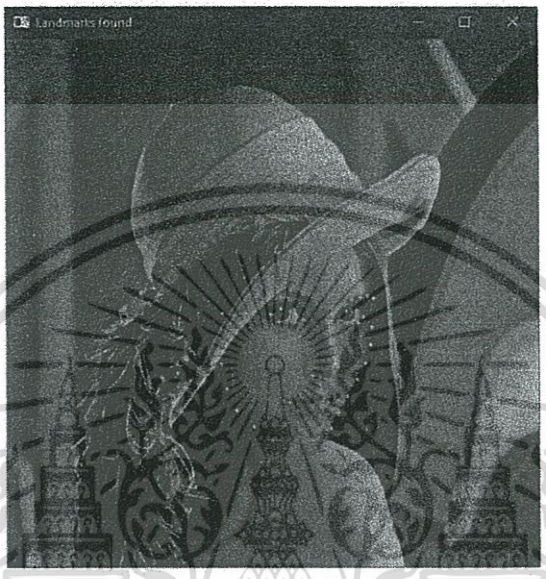
ตัวอย่างของ Dlib ในด้านของการประมานผลภาพ (Image Processing)

- การอ่านและเขียนรูปภาพประเภทต่างๆ
- การแปลง color space สำหรับ pixel หลากหลายประเภท
- การประมานผลภาพทั่วไปเช่นการหาขอบภาพ หรือ การเปลี่ยนโครงสร้างของภาพ
- ชุดคำสั่งของการทำ SURF, HOG, และ FHOG

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- การตรวจจับวัตถุในภาพและการวางท่าทางต่างๆ รวมถึง ใบหน้าและการวางท่าทางของใบหน้า และส่วนประกอบต่างๆของใบหน้า 68 จุด (facial landmarks)
- การจดจำใบหน้าความละเอียดสูง High quality face recognition

ตัวอย่างผลลัพธ์การทำงานของ Dlib แสดงดังรูปที่ 2.9



รูปที่ 2.9 ตัวอย่างการใช้ Dlib เพื่อหา facial landmark 68 จุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

วิธีการดำเนินงานวิจัย

ผู้จัดทำได้แบ่งวิธีการดำเนินงานออกเป็น 3 ส่วนดังนี้

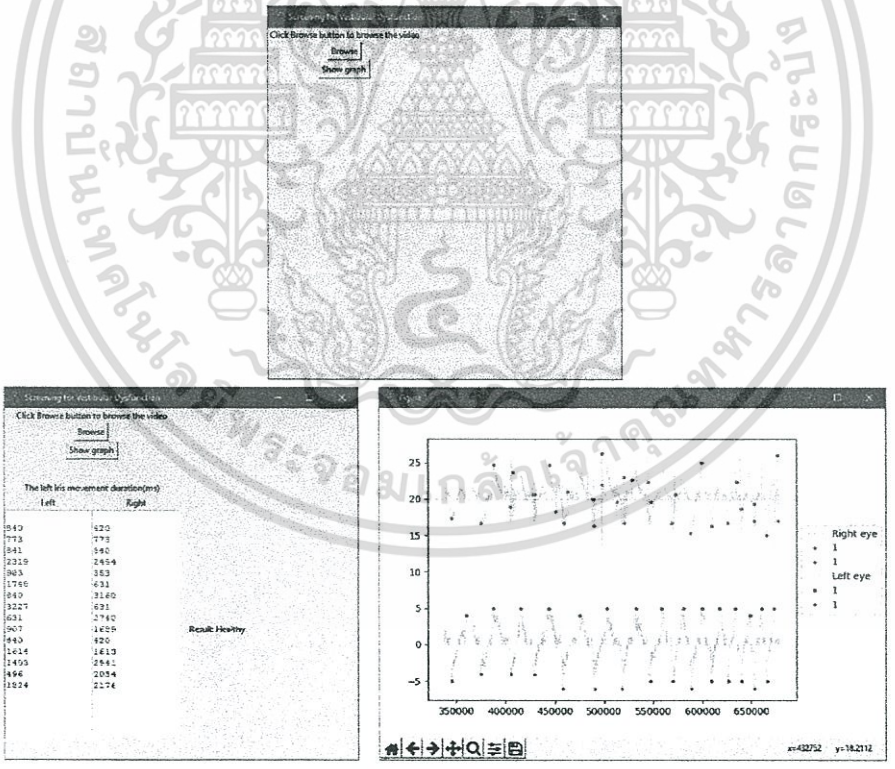
3.1 การออกแบบโปรแกรม

3.2 กรอบแนวคิดในการทำวิจัย

3.3 การทดสอบโปรแกรม

3.1 การออกแบบโปรแกรม

การสร้างโปรแกรมฯใช้ภาษา Python ด้วยโปรแกรม PyCharm Education ซึ่งเป็น Open source ใช้ library หลักคือ OpenCV ในการประมวณผลภาพ และ Dlib เพื่อหาตำแหน่งต่างๆบนใบหน้า (facial landmark) หน้าต่าง GUI ของโปรแกรมแสดงในรูปที่ 3.1



รูปที่ 3.1 (บน) หน้าต่างเริ่มต้นของโปรแกรมฯ (ล่าง) หน้าต่างของโปรแกรมเมื่อกดปุ่ม show graph

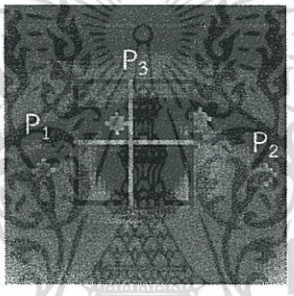
โปรแกรมฯเริ่มทำงานโดยให้ผู้ใช้งานกดปุ่ม Browse แล้วเลือกคลิปวิดีโอ จากนั้นโปรแกรมจะแสดงภาพเฟรมหนึ่งของวิดีโอให้ผู้ใช้งานกำหนด template ของตาดำข้างซ้ายและข้างขวาโดยการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

crop ขอบเขตภาพที่ต้องการ เพื่อนำไปใช้ในการทำ template matching ต่อไป โปรแกรมจะทำการประมวนผลในทุกเฟรมของวิดีโอเพื่อตรวจหาใบหน้าและจุดต่างๆบนใบหน้าตามลำดับ แล้วทำการกำหนด ROI (Region of Interest) บริเวณของดวงตาทั้งสองข้าง จากนั้นทำ template matching ในบริเวณ ROI ของตาแต่ละข้าง เพื่อหาตำแหน่งของดวงตา จากนั้นคำนวณตำแหน่งของตาดำ (D) เทียบกับจุดอ้างอิงคือหัวตาและหางตา เป็นไปตามสมการที่ 1 โดยการคำนวณพิจารนาเฉพาะค่าพิกัดในแกน x

$$D = \frac{P_1+P_2}{2} - P_3 \quad (1)$$

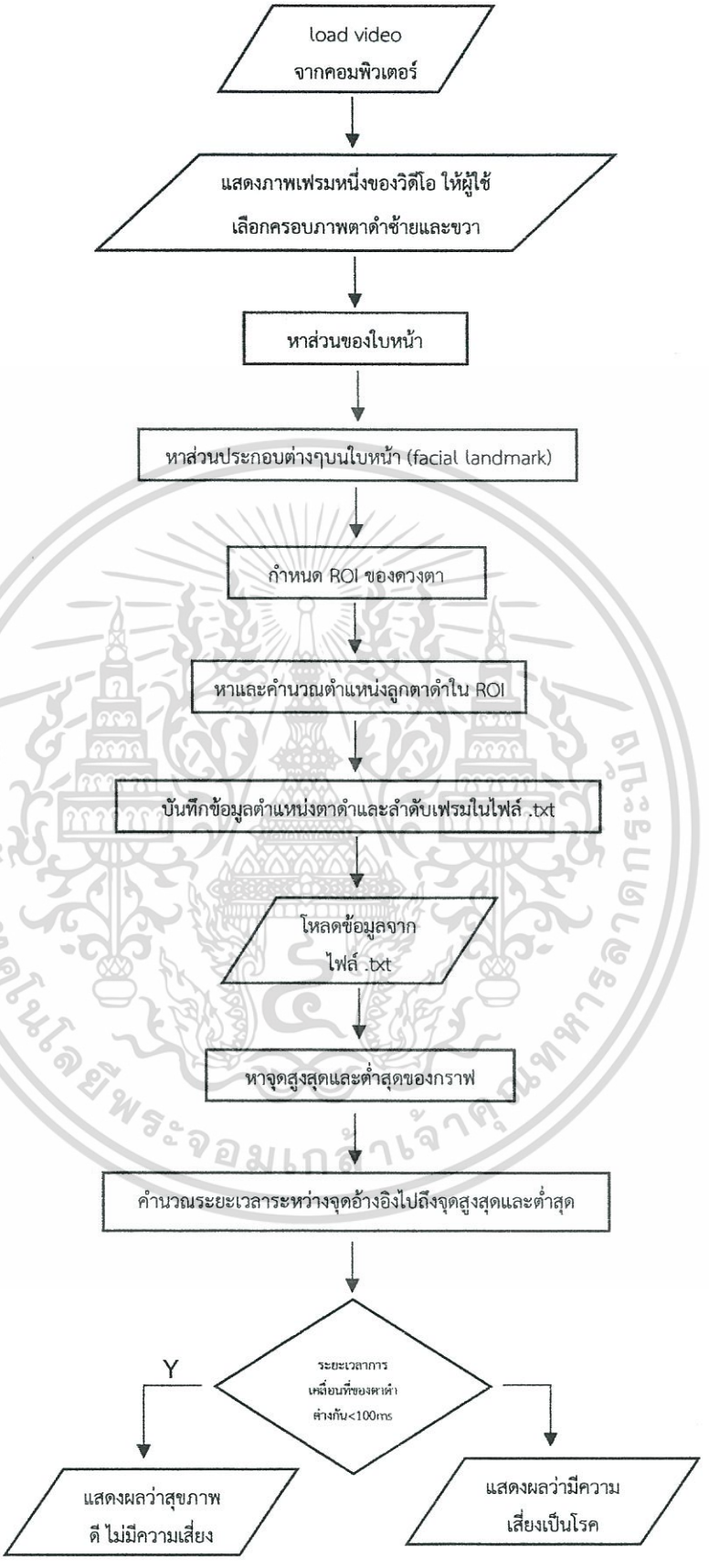
โดยที่ P_1 และ P_2 คือจุดของหัวตาและหางตา, P_3 คือจุดกึ่งกลางสี่เหลี่ยม (ตำแหน่งจุดกึ่งกลางของดวงตาที่ตรวจจับได้) จุดต่างๆแสดงในรูปที่ 3.2



รูปที่ 3.2 ROI บริเวณดวงตา และจุดอ้างอิงที่ใช้ในการคำนวณ

ตำแหน่งของตาดำและลำดับที่ของเฟรมนั้นจะถูกบันทึกไว้ในไฟล์ .txt หลังจากนั้นผู้ใช้งานต้องกดปุ่ม show graph เพื่อให้โปรแกรมแสดงผลเป็นกราฟโดยจะโหลดข้อมูลจากไฟล์ .txt และทำการหาจุดสูงสุดและจุดต่ำสุดของกราฟ ซึ่งจะเป็นตำแหน่งของลูกตาขณะหันศีรษะ จากนั้นคำนวณหาระยะเวลาที่ตาดำเคลื่อนที่จากเริ่มต้น(จุดกึ่งกลางดวงตา)ไปยังทางซ้ายขวา(จุดสูงสุด,ต่ำสุดของกราฟ) และเปรียบเทียบระยะเวลาการเคลื่อนที่ของตาดำไปทางซ้ายและขวาว่ามีความใกล้เคียงกันหรือไม่ ถ้ามีค่าใกล้เคียงกันจะแสดงผลว่าไม่มีความเสี่ยงเป็นโรคฯ ถ้าไม่เช่นนั้นจะแสดงผลว่ามีความเสี่ยงเป็นโรคฯ โดยกระบวนการทำงานทั้งหมดของโปรแกรมฯนั้นเป็นดังรูปที่ 3.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.3 แผนภาพกระบวนการทำงานของโปรแกรมฯ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 กรอบแนวคิดในการทำวิจัย

โปรแกรมที่สร้างขึ้นมานั้นสามารถใช้ตรวจคัดกรองโรค Vestibular dysfunction ชนิดที่ vestibular system เกิดความเสียหายเพียงข้างเดียวเท่านั้น ถ้าเกิดความเสียหายทั้งสองข้างจะไม่สามารถแยกความแตกต่างระหว่างระยะเวลาการเคลื่อนที่ของตาไปทางซ้ายกับขวาได้

3.3 การทดสอบโปรแกรม

ทดสอบการทำงานของโปรแกรมฯโดยใช้วิดีโอของผู้ทดสอบที่สุขภาพดี โดยบันทึกจากกล้องหลังของโทรศัพท์ Sony Xperia XZ premium ความละเอียด 1280x720 frame rate 120 เฟรมต่อวินาที

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

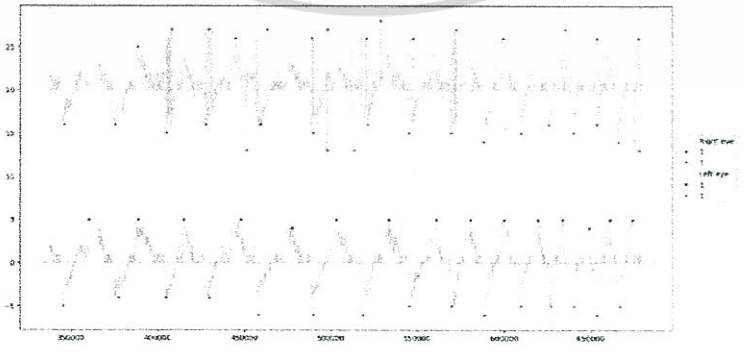
ผลการทดลอง

4.1 ผลการทดลอง

การทดสอบการทำงานของโปรแกรมนั้นผู้ทดลองได้ทำโดยใช้วิดีโอที่ถ่ายด้วยกล้องโทรศัพท์มือถือ Sony Xperia XZ premium ความละเอียด 1280x720 120fps ถ่ายภาพขณะทำการหันศีรษะไปทางซ้ายและขวาอย่างรวดเร็วจำนวน 15 ครั้ง การทำงานของโปรแกรมฯเป็นไปตามรูปที่ 4.1

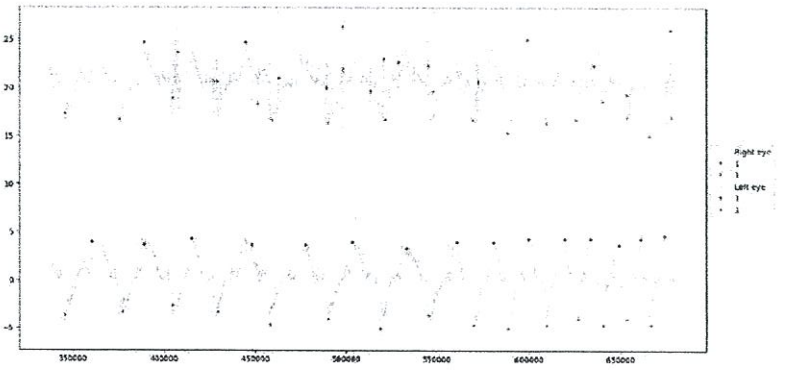
รูปที่ 4.1 โปรแกรมขณะทำการตรวจหาใบหน้าและตาดำ

กราฟตำแหน่งการเคลื่อนที่ของดวงตาแสดงในรูปที่ 4.2 และกราฟที่ถูกปรับปรุงด้วย filter moving average แบบ 3 จุดแสดงในรูปที่ 4.3



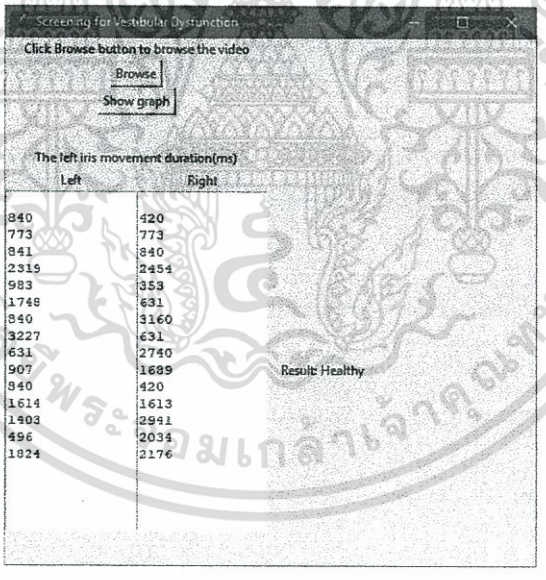
รูปที่ 4.2 กราฟตำแหน่งของตาดำซ้ายและขวา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.3 กราฟตำแหน่งของตาดำซ้ายและขวา (หลังจากทำ moving average แบบ 3 จุด)

จากกราฟของตาขวาจะเห็นว่าผลลัพธ์ที่ออกมายังมีความผิดพลาดอยู่บ้างทั้งในการหาจุดสูงสุดต่ำสุดและการหาตำแหน่งของตาดำเนื่องจากการตรวจจับตำแหน่งของใบหน้าหรือตาดำเองผิดพลาด ซึ่งในการวิเคราะห์ผลนั้นสามารถใช้ผลลัพธ์การเคลื่อนที่ของตาดำข้างใดข้างหนึ่งได้เนื่องจากตาทั้งสองข้างจะเคลื่อนที่พร้อมกันและความเร็วเท่ากัน ผลลัพธ์ระยะเวลาการเคลื่อนที่ของดวงตานั้นแสดงในหน้าต่างของโปรแกรม รูปที่ 4.4



รูปที่ 4.4 หน้าต่างของโปรแกรมฯเมื่อกด show graph

ผลลัพธ์ของโปรแกรมฯ ยังมีจุดที่ต้องปรับปรุงและพัฒนาในอนาคต นอกจากนั้นจะทำการทดสอบกับผู้ป่วย และมีการพัฒนาโปรแกรมต่อไปเพื่อใช้ในสมาร์ทโฟนต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
สรุปผลและปัญหาที่เกิดขึ้น

5.1 สรุปผลและอภิปรายผล

โรค Vestibular Dysfunction เป็นหนึ่งในสาเหตุที่ก่อให้เกิดอาการวิงเวียนศีรษะแบบบ้านหมุน ในปัจจุบันโรคนี้ไม่ได้เป็นที่รู้จักอย่างกว้างขวาง การวินิจฉัยโรคต้องใช้แพทย์ผู้เชี่ยวชาญและมีประสบการณ์ โดยแพทย์และผู้ป่วยนั่งตัวตรงหัวหน้าเข้าหากัน ผู้ป่วยจ้องที่จุดๆหนึ่งตลอดเวลา (ส่วนมากนิยมเป็นปลายจมูกของแพทย์) จากนั้นแพทย์จะทำการจับศีรษะผู้ป่วยหัน (ประมาณ 15 องศา) อย่างรวดเร็วไปทางซ้ายและขวาแล้วดูระยะเวลาการเคลื่อนที่ของลูกตาเพื่อกลับมามองที่จมูกเหมือนเดิม ผู้ที่มีความผิดปกตินั้นลูกตาจะไม่กลับมาจ้องที่เดิมโดยอัตโนมัติหรือรีเฟล็กซ์ และต้องบังคับกล้ามเนื้อให้กรอกตามาด้วยตนเอง เพราะฉะนั้นจะใช้ระยะเวลามากกว่าการใช้รีเฟล็กซ์ แพทย์จะสามารถเปรียบเทียบระหว่างการเคลื่อนที่ของลูกตาไปทางซ้ายและขวาได้

การสร้างโปรแกรมนั้นใช้ภาษา Python และ library หลักคือ OpenCV และ Dlib สำหรับการประมวนผลภาพต่างๆและการหาตำแหน่งบนใบหน้า 68 จุด (facial landmark) ตามลำดับ การทำงานของโปรแกรมใช้วิดีโอที่มี 120 เฟรมต่อวินาที เริ่มต้นผู้ใช้งานต้องทำการ crop ส่วนของตาดำทั้งซ้ายและขวาก่อนเพื่อโปรแกรมจะนำไปใช้หาตำแหน่งตาดำต่อไป ในทุกเฟรมของวิดีโอโปรแกรมจะทำการตรวจหาใบหน้า ส่วนต่างๆของใบหน้า กำหนด ROI ของดวงตา และหาตำแหน่งของดวงตาตามลำดับ จากนั้นนำค่าตำแหน่งไปคำนวณหาตำแหน่งของตาดำเทียบกับจุดอ้างอิงคือหัวและหางตา การแสดงผลเป็นกราฟและการวิเคราะห์ผลจะทำหลังจากที่ได้ข้อมูลจากทุกเฟรม การวิเคราะห์ความเสี่ยงเป็นโรคนั้นวัดจากระยะเวลาการเคลื่อนที่ของดวงตา จากจุดอ้างอิงไปยังจุดสูงสุดหรือต่ำสุดของกราฟ แล้วเปรียบเทียบระยะเวลาการกรอกตาไปทางซ้ายกับขวาว่าใกล้เคียงกันหรือไม่ หากต่างกันเกิน 100 ms [2] จะถือว่ามีความเสี่ยงเป็นโรค

การทดสอบการทำงานของโปรแกรมทำโดยใช้วิดีโอของผู้ทดสอบที่สุขภาพดี ผลลัพธ์ของโปรแกรมค่อนข้างน่าพอใจ แต่ยังมีส่วนที่ผิดพลาดและความปรับปรุงอยู่มากพอสมควร ซึ่งผู้จัดทำจะพัฒนาต่อไปในระดับปริญญาโท และโปรแกรมจะถูกพัฒนาต่อไปในอนาคตเพื่อทดสอบกับผู้ป่วยจริงและสามารถใช้งานในสมาร์ทโฟน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2 ปัญหาและแนวทางแก้ไข

จากการพัฒนาโปรแกรมฯการทำงานของโปรแกรมค่อนข้างดี แต่ยังประสบปัญหาและมีข้อผิดพลาดอยู่ที่ยังทำให้โปรแกรมฯยังไม่สมบูรณ์ ดังนี้

5.2.1 การทำงานของโปรแกรมค่อยข้างช้า เนื่องจาก การใช้ Haar Cascade Classifier ตรวจจับใบหน้าใช้เวลาประมวนผลนานเนื่องจากต้องสแกนทั่วทั้งภาพ, ขนาดภาพของแต่ละเฟรมที่ใหญ่, จำนวนเฟรมทั้งหมดของวิดีโอที่มาก

5.2.2 การตรวจจับลูกตายังมีความแม่นยำประมาณ 90% ทำให้ข้อมูลและกราฟที่ออกมามีความผิดพลาดได้

5.2.3 แสงและเงามีผลต่อการตรวจจับใบหน้า อาจทำให้ตรวจจับผิดพลาดหรือตรวจไม่พบใบหน้าได้

5.2.4 โปรแกรมยังมีส่วนที่ต้องอาศัยผู้ใช้งานควบคุม คือการเลือก crop ตำแหน่งตาดำเพื่อทำ Template matching

5.2.5 การหาจุดสูงสุดและต่ำสุดยังมีความผิดพลาดอยู่ ทั้งเนื่องจากการตรวจจับใบหน้าผิดพลาด, ตำแหน่งตาดำผิดพลาด และการตรวจหาจุดสูงสุดผิดพลาดเอง

5.2.6 ปัญหาอาจเกิดจากการทดลอง เพราะผู้ทดลองทำการทดลองสะบัดหน้าเองไม่ได้ให้แพทย์ทดสอบเหมือนกับที่ทดสอบผู้ป่วยจริง อาจทำให้ความเร็วและองศาของการทำแต่ละครั้งนั้นไม่เท่ากันทำให้ส่งผลถึงผลลัพธ์ที่ผิดพลาด

ซึ่งปัญหาข้างต้นที่กล่าวมาผู้จัดทำได้นำเสนอวิธีแก้ไขบางส่วนดังนี้

ทำการศึกษาเพิ่มเติมเกี่ยวกับโรคฯและความเร็ว, ระยะเวลาการเคลื่อนที่ของดวงตาเพื่อนำมาเป็นเกณฑ์ที่ถูกต้องในการวิเคราะห์ความเสี่ยงของการเป็นโรคฯ ศึกษาวิธีอื่นๆในการตรวจหาใบหน้าแทน Haar Cascade Classifier และการตรวจจับตาดำแทน Template matching เพื่อให้โปรแกรมทำงานโดยอัตโนมัติซึ่งผู้ใช้งานไม่ต้องเลือกตำแหน่งตาดำตอนเริ่มต้น ใช้วิดีโอการทำการทดสอบขณะแพทย์ทดสอบจริงโดยใช้ทั้งผู้ทดสอบที่สุขภาพดีและผู้ป่วย ซึ่งทั้งหมดจะเป็นการพัฒนาในระดับปริญญาโทต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารอ้างอิง

[1] Vestibular system, https://th.wikipedia.org/wiki/Vestibular_system. Last access Oct 17, 2017

[2] Georgios Mantokoudis and authers (2014), Quantifying the Vestibulo-Ocular Reflex with Video-Oculography: Nature and Frequency of Artifacts, Audiol Neurotol 2015;20:39–50, DOI: 10.1159/000362780

[3] HIT for VOR, แปลจาก: https://bpac.org.nz/BPJ/2012/september/vertigo.aspx. Last access Apr 12, 2018

[4] การทรงตัว, www.sso.go.th/sites/default/files/WYSIWYG%20Web%20Builder/sso.html/Chapter_7_440.html. Last access Oct 17, 2017

[5] โรคเส้นประสาทการทรงตัวอักเสบ, https://www.honestdocs.co/vestibular-neuritis. Last access Oct 17, 2017

[6] เวียนศีรษะ (Dizziness), http://haamor.com/th/เวียนศีรษะ. Last access Oct 17, 2017

[7] เวียนศีรษะบ้านหมุน, https://www.bumrungrad.com/healthspot/March2015/vertigo-treatment. Last access Oct 17, 2017

[8] รู้จักกับภาษา Python, http://bbee.exteen.com/20080313/python. Last access Oct 17, 2017

[9] ภาษาไพทอน, https://th.wikipedia.org/wiki/ภาษาไพทอน. Last access Oct 17, 2017

[10] รศ.ดร. ชูชาติ ปิณฑวิรุจน์. (2552). การประมาณผลภาพดิจิตอลด้วย C++ DIGITAL IMAGE PORCESSING WITH C++. พิมพ์ครั้งที่1. กรุงเทพฯ: แผนกผลิตจำราและสื่อการสอน คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้า เจ้าคุณทหาร ลาดกระบัง.

[11] OpenCV, แปลจาก: https://en.wikipedia.org/wiki/OpenCV. Last access Oct 20, 2017

[12] Template matching, แปลจาก: https://docs.opencv.org/trunk/d4/dc6/tutorial_py_template_matching.html. Last access Apr 12, 2018

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารอ้างอิง(ต่อ)

[13] Template matching, แปลจาก: https://docs.opencv.org/master/de/da9/tutorial_template_matching.html. Last access May 7, 2018

[14] Dlib, แปลจาก: http://dlib.net/. Last access Nov 22, 2017

[15] Dlib, แปลจาก: http://dlib.net/intro.html, Last access May 9, 2018

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

Code ของโปรแกรม

```
from tkinter import *
from tkinter import filedialog
import cv2
import dlib
import numpy as np
import peakutils.peak
import matplotlib.pyplot as plt
from tkinter import messagebox
import pandas as pd
import peakutils

def browse():
    bv = filedialog.askopenfilename(filetypes = (("MOV, mp4",
"*.MOV;*.mp4"),("All files", "*.*") ))
    cascPath = "haarcascade_frontalface_default.xml"
    PREDICTOR_PATH = "shape_predictor_68_face_landmarks.dat"
    FACELINE_POINTS = list(range(0, 68))
    # Create the haar cascade
    faceCascade = cv2.CascadeClassifier(cascPath)
    predictor = dlib.shape_predictor(PREDICTOR_PATH)
    cap = cv2.VideoCapture(bv)
    fps = int(cap.get(cv2.CAP_PROP_FPS))
    tof = int(cap.get(cv2.CAP_PROP_FRAME_COUNT))
    #template = cv2.imread('frame.jpg',0)
    threshold = 0.4
    print(fps,tof)
    # Read image
    im = cv2.imread("cap.jpg")
    # Select ROI
    messagebox.showinfo("Crop a template", " --> Crop the 'right'
iris then press 'Enter' ot 'Space Bar' --> ")
    r = cv2.selectROI(im)
    # Crop image
    templateR = im[int(r[1]):int(r[1]+r[3]),
int(r[0]):int(r[0]+r[2])]
    # Display cropped image
    cv2.imshow('R',templateR)
    templateR = cv2.cvtColor(templateR, cv2.COLOR_BGR2GRAY)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    im = cv2.imread("cap.jpg")
    messagebox.showinfo("Crop a template", " <-- Crop the 'left' iris
then press 'Enter' ot 'Space Bar' <-- ")
    # Select ROI
    r = cv2.selectROI(im)
    # Crop image
    templateL = im[int(r[1]):int(r[1]+r[3]),
int(r[0]):int(r[0]+r[2])]
    # Display cropped image
    cv2.imshow('L',templateL)
    templateL = cv2.cvtColor(templateL, cv2.COLOR_BGR2GRAY)
    text_file = open("OutputR.txt", "w")
    text_file.write("0,0\n")
    text_file = open("OutputL.txt", "w")
    text_file.write("0,0\n")
    text_file.close()

    while(cap.isOpened()):
        ret, frame = cap.read()
        Ttime = int(cap.get(cv2.CAP_PROP_POS_MSEC))
        print(Ttime)
        Tsec = Ttime*1000/fps
        if Ttime>400000:
            gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)

            image=frame

            faces = faceCascade.detectMultiScale(gray, 1.3,5)

            if len(faces) > 0:
                # Draw a rectangle around the faces
                for (x, y, w, h) in faces:
                    cv2.rectangle(image, (x, y), (x + w, y + h), (0,
255, 0), 2)

                    dlib_rect = dlib.rectangle(int(x), int(y), int(x
+ w), int(y + h))

                    landmarks = np.matrix([[p.x, p.y]for p in
predictor(image, dlib_rect).parts()])
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                    landmarks_display = landmarks[FACELINE_POINTS]

                    P=[0,0,0,0,0,0,0,0]
                    for idx, point in enumerate(landmarks_display):
                        pos = (point[0, 0], point[0, 1])
                        idx+=1
                        if (37<=idx<=48):
                            cv2.circle(frame, pos, 2, color=(255,
125, 255), thickness=-1)
                            if (idx==37):
                                P[0] = point[0,0]
                                P[1] = point[0,1]
                            elif (idx==40):
                                P[2] = point[0,0]
                                P[3] = point[0,1]
                            elif (idx==43):
                                P[4] = point[0,0]
                                P[5] = point[0,1]
                            elif (idx==46):
                                P[6] = point[0,0]
                                P[7] = point[0,0]
                            idx-=1
                    r = (P[1]+((P[2]-P[0])/2))
                    o = (P[1]-((P[2]-P[0])/2))
                    roi = frame[int(o):int(r), P[0]:P[2]]

                    grayL = gray[int(o):int(r), P[0]:P[2]]
                    ww, hh = templateL.shape[::-1]

                    res =
cv2.matchTemplate(grayL,templateL,cv2.TM_CCORR_NORMED)
                    if len(res)>0:
                        min_val, max_val, min_loc, max_loc =
cv2.minMaxLoc(res)
                        top_left = max_loc
                        bottom_right = (top_left[0]+ ww, top_left[1]+ hh)
                        cv2.rectangle(roi,top_left, bottom_right, 255, 1)
                        cv2.line(roi, (int((top_left[0]+ww/2)),
int((top_left[1]))),(int((top_left[0]+ww/2)),
int(top_left[1]+hh)),(0,255,255),1)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                    cv2.line(roi, (int((top_left[0])),
int((top_left[1]+hh/2))),(int((top_left[0]+ww)),
int(top_left[1]+hh/2)),(0,255,255),1)
                    Hor = int(top_left[0]+ww/2)-(P[2]-P[0])/2
                    text_file = open("OutputL.txt", "a")
                    text_file.write("%d,%d\n"%(Tsec,Hor))
                    text_file.close()

                roiR = frame[int(P[5]-((P[6]-
P[4])/2)):int(P[5]+((P[6]-P[4])/2)), P[4]:P[6]]
                grayR = gray[int(P[5]-((P[6]-
P[4])/2)):int(P[5]+((P[6]-P[4])/2)), P[4]:P[6]]

                res =
cv2.matchTemplate(grayR,templateR,cv2.TM_CCORR_NORMED)
                if len(res)>0:
                    min_val, max_val, min_loc, max_loc =
cv2.minMaxLoc(res)
                    top_left = max_loc
                    bottom_right = (top_left[0] + ww, top_left[1] +
hh)
                    cv2.rectangle(roiR,top_left, bottom_right,
(0,0,255), 1)
                    cv2.line(roiR, (int((top_left[0]+ww/2)),
int((top_left[1]))),(int((top_left[0]+ww/2)),
int(top_left[1]+hh)),(0,255,255),1)
                    cv2.line(roiR, (int((top_left[0])),
int((top_left[1]+hh/2))),(int((top_left[0]+ww)),
int(top_left[1]+hh/2)),(0,255,255),1)
                    Hor = int(top_left[0]+ww/2)-(P[6]-P[4])/2
                    text_file = open("OutputR.txt", "a")
                    text_file.write("%d,%d\n"%(Tsec,Hor))
                    text_file.close()

            cv2.imshow("Landmarks found", image)
            if cv2.waitKey(1) & 0xFF == ord('q'):
                break

    cap.release()
    cv2.destroyAllWindows()

def putinfo():
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    text1.grid(row=6, column=1)
    text2.grid(row=6, column=2)
    labelL.grid(row=5, column=1)
    labelR.grid(row=5, column=2)
    labelM.grid(row=4, column=1,columnspan=2)

    offset = 20
    estimated_data = pd.read_csv("OutputR.txt", header=None)
    col1 = estimated_data[:][0] # First column data
    col2 = estimated_data[:][1] # Second column data
    col3 = -1*col2
    index1 = peakutils.indexes(col2, thres=0.75, min_dist=100)
    index2 = peakutils.indexes(col3, thres=0.75, min_dist=100)
    col2 = pd.rolling_mean(col2, 3)
    #print(index1)
    #print(index2)
    plt.plot(col1,col2+offset,'y', lw=0.4, alpha=0.4, label='Right
eye')
    plt.plot(col1[index1],col2[index1]+offset, marker="o", ls="",
ms=3 )
    plt.plot(col1[index2],col2[index2]+offset, marker="o", ls="",
ms=3 )
    estimated_data = pd.read_csv("OutputL.txt", header=None)
    col1 = estimated_data[:][0] # First column data
    col2 = estimated_data[:][1] # Second column data
    col3 = -1*col2
    index = peakutils.indexes(col2, thres=0.8, min_dist=100)
    indexL = peakutils.indexes(col3, thres=0.8, min_dist=100)

    DR = []
    DL = []

    for i in range(0, len(index)):
        DR.append(col1[index[i]])
    print(DR)
    for i in range(0, len(indexL)):
        DL.append(col1[indexL[i]])
    print(DL)

    z=[]
    for i in range(0,len(col2)):
        if col2[i]==0 or col2[i]==1:
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        z.append(col1[i])
print(len(z))

R=[]
for i in reversed(range(0,len(DR))):
    for j in reversed(range(0,len(z))):
        if z[j]<DR[i]:
            print(z[j])
            R.append(DR[i]-z[j])
            break
print(R)
print(len(R))
L=[]
for i in reversed(range(0,len(DL))):
    for j in reversed(range(0,len(z))):
        if z[j]<DL[i]:
            print(z[j])
            L.append(DL[i]-z[j])
            break
print(L)
print(len(L))

text1.delete('1.0', END)
text2.delete('1.0', END)
for i in range(0,len(L)):
    text1.insert(1.0,L[i])
    text1.insert(1.0,'\n')
for i in range(0,len(R)):
    text2.insert(1.0,R[i])
    text2.insert(1.0,'\n')
diff=[]
de=0
he=0
for i in range(0,len(R)):
    diff.append(R[i]-L[i])
print(diff)
for i in range(0,len(diff)):
    if diff[i]>400:
        de+=1
    if diff[i]<=400:
        he+=1
print (de,he)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    if he>de:
        helthy = Label(frame1, text = "\n   Result: Healthy")
        helthy.grid(row=6, column=3,columnspan=2)
    else:
        Risk = Label(frame1, text = "   Result: Risk")
        Risk.grid(row=6, column=3,columnspan=2)

    '''
    for i in range(0, len(indexL)):
        DL[i]=indexL[i]
        text2.insert(1.0,DL[i])
        text2.insert(1.0,'\n')
    '''
    col2 = pd.rolling_mean(col2, 3)
    plt.plot(col1,col2, lw=0.4, alpha=0.4, label='Left eye' )
    plt.plot(col1[index],col2[index], marker="o", ls="", ms=3 )
    plt.plot(col1[indexL],col2[indexL], marker="o", ls="", ms=3 )
    plt.axhline(y=0,lw=0.4, alpha=0.4)
    plt.axhline(y=offset,lw=0.4, alpha=0.4)
    plt.legend(loc='center left', bbox_to_anchor=(1, 0.5))

    plt.show()

root= Tk()
root.title("Screening for Vestibular Dysfunction")
root.geometry("500x500")
frame1=Frame(root)
frame1.grid()
label1 = Label(frame1, text = "Click Browse button to browse the
video")
label1.grid(row=1, column=1,columnspan=2)
button1 = Button(frame1, text = "Browse",command=browse)
button1.grid(row=2, column=1,columnspan=2)
button2 = Button(frame1, text = "Show graph",command=putinfo)
button2.grid(row=3, column=1,columnspan=2)
text1 = Text(frame1, width = 15, height = 20)
text2 = Text(frame1, width = 15, height = 20)
labelM = Label(frame1, text = "\n\nThe left iris movement
duration(ms)")
labelL = Label(frame1, text = "Left")
labelR = Label(frame1, text = "Right")
root.mainloop()
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้