



การศึกษาผลของ Window function ต่อความแม่นยำในการจำแนกเสียงเครื่องดนตรีด้วย FFT และ K-NN Classifier
Study of effect of window function on musical instrument classification with FFT and K-NN Classifier

สาขา วิทยาการคอมพิวเตอร์

รายงานฉบับสมบูรณ์

เสนอต่อ

ศูนย์เทคโนโลยีอิเล็กทรอนิกส์และคอมพิวเตอร์แห่งชาติ

สำนักงานพัฒนาวิทยาศาสตร์และเทคโนโลยีแห่งชาติ

กระทรวงวิทยาศาสตร์และเทคโนโลยี

ได้รับทุนอุดหนุนโครงการวิจัย พัฒนาและวิศวกรรม

โครงการ “การประกวดโครงการของนักวิทยาศาสตร์รุ่นเยาว์ ครั้งที่ 19”

ประจำปีงบประมาณ 2559

โดย

นายเชมรัฐ บุญยะผลึก

นายจิตรภาณุ อัครพิชญโชติ

นายพุฒิพัฒน์ สิริเศรษฐภักดี

นายอภิรักษ์ เงินมูล

โรงเรียนสามเสนวิทยาลัย

ข้อมูลผู้พัฒนาโครงการ

หัวหน้าทีม

ชื่อ-นามสกุล นายเขมรัฐ บุญยะผลึก

วัน/เดือน/ปีเกิด 3 มีนาคม 2542 ระดับการศึกษา มัธยมศึกษาปีที่ 6

สถานศึกษา โรงเรียนสามเสนวิทยาลัย

ที่อยู่ตามทะเบียนบ้าน 89/47 หมู่ 6 ถนนกรุงนนท์-จตุรนต์ ตำบลศาลากลาง อำเภอบางกรวย นนทบุรี

มือถือ 084-433-5633 E-mail korla.march@gmail.com

1) ผู้พัฒนาโครงการ

ชื่อ-นามสกุล นายจิตรภาณุ อัสวพิชญโชติ

วัน/เดือน/ปีเกิด 4 ตุลาคม 2541 ระดับการศึกษา มัธยมศึกษาปีที่ 6

สถานศึกษา โรงเรียนสามเสนวิทยาลัย

ที่อยู่ตามทะเบียนบ้าน 169 ซ.ปทุมทิพย์ ถนนกรุงเทพฯ-นนท์ แขวงบางซื่อ เขตบางซื่อ กรุงเทพมหานคร

มือถือ 086-398-8267 E-mail jewgod@hotmail.com

2) ผู้พัฒนาโครงการ

ชื่อ-นามสกุล นายวุฒิพัฒน์ สิริเศรษฐภักดิ์

วัน/เดือน/ปีเกิด 21 มีนาคม 2542 ระดับการศึกษา มัธยมศึกษาปีที่ 6

สถานศึกษา โรงเรียนสามเสนวิทยาลัย

ที่อยู่ตามทะเบียนบ้าน 170/3 ถนนประชากรราษฎร์บำเพ็ญ แขวงห้วยขวาง เขตห้วยขวาง กรุงเทพมหานคร

มือถือ 099-465-4991 E-mail taitai.12486-@live.com

อาจารย์ที่ปรึกษา

ชื่อ-นามสกุล นายอนันต์ เงินมูล

สังกัด/สถาบัน โรงเรียนสามเสนวิทยาลัย

สถานที่ติดต่อ 132/11 ถนนพระราม 6 สามเสนใน เขตพญาไท กรุงเทพฯ

โทรศัพท์ 0-2278-2997 มือถือ 081-840-0779 E-mail mightymouse9669@hotmail.com

สารบัญ

หัวข้อ	หน้า
บทคัดย่อ	1
วัตถุประสงค์	2
ตัวแปรในการทดลอง	2
สมมุติฐาน	2
ซอฟต์แวร์/อุปกรณ์ที่ใช้ในการทดลอง	2
ขั้นตอนการดำเนินงาน	2
ขั้นตอนในการทำงานของอัลกอริทึมจำแนกเสียง	6
ผลการทดลอง	7
วิเคราะห์ผลการทดลอง	17
อภิปรายผลการทดลองและข้อเสนอแนะ	18
สรุปผลการทดลอง	18
กิตติกรรมประกาศ	19
บรรณานุกรม	20

บทคัดย่อ

ปัจจุบัน การจำแนกเสียงเป็นออกเป็นประเภทต่างๆ โดยการใช้คอมพิวเตอร์ได้รับความนิยมสูงมาก เนื่องจากสามารถนำไปประยุกต์ใช้หลากหลาย ยกตัวอย่างเช่น การแปลงเสียงออกมาเป็นข้อความ, การยืนยันตัวตนด้วยเสียง หรือการนำเสียงของผู้ร้ายไปหาผู้ต้องสงสัยในคดีอาชญากรรม เป็นต้น ซึ่งสิ่งเหล่านี้ ต่างต้องการอัลกอริทึมที่สามารถจำแนกเสียงออกเป็นกลุ่มต่างๆ ได้อย่างรวดเร็วและแม่นยำด้วยกันทั้งสิ้น ดังนั้น พวกเราจึงสนใจพัฒนาอัลกอริทึมในการจำแนกเสียงของเครื่องดนตรี 3 ชนิด ได้แก่ กีตาร์ เปียโน และ ไวโอลิน โดยที่จะนำ Window functions ชนิดต่างๆ ได้แก่ Rectangular window, Triangular window, Hanning window, Hamming window, Blackman-Harris window และ Flat top window มาใช้ร่วมกับการจำแนกเสียงด้วย Fast Fourier Transform และ k-Nearest Neighbors (k-NN) เพื่อเพิ่มความแม่นยำในการจำแนกเสียง ซึ่งสำหรับหา Nearest Neighbors เราก็ได้ทดลองแบบทั้งใช้ Euclidean Distance และ Manhattan Distance ในการหา นอกจากนี้ยังได้ทดลองกับค่า k (ใน k-NN) ตั้งแต่ 1-5 โดยได้พัฒนาโปรแกรมในภาษา C++ และ Node.js ขึ้น เพื่อทดลองสมมุติฐาน และทดสอบความถูกต้องของอัลกอริทึม ซึ่งเราได้ใช้เสียงตัวอย่างจากเครื่องดนตรี 3 ชนิด ชนิดละ 50 เสียง รวม 150 เสียง มาทดสอบ โดยการทดสอบจะถูกจัดขึ้นในสภาพแวดล้อมที่ต่างกัน 5 ชุด คือ มีเสียงรบกวน 0%, 10%, 25%, 50%, 75% การภายหลังการทดสอบ เราสามารถ สรุปผลการทดลองได้ว่าการทดลองนั้นเป็นไปตามสมมุติฐานที่ตั้งไว้ นั่นคือการใช้ Window function ช่วยเพิ่มความแม่นยำของการจำแนกเสียงได้จริง และในกรณีที่ใช้ $k=1$, Euclidean Distance และ Flat top windowing จะมีอัตราการตอบถูกที่สูงที่สุดในสภาพแวดล้อมที่ไม่มีเสียงรบกวนถึงร้อยละ 96.00 และการใช้ค่า k เป็น 1,3,5 จะมีประสิทธิภาพในการแยกแยะเสียงสูงกว่ากรณีที่ค่า k เป็น 2,4

วัตถุประสงค์

การจำแนกเสียงเป็นออกเป็นประเภทต่างๆ โดยการใช้คอมพิวเตอร์นั้น มีความนิยมสูงมาก เนื่องจากสามารถนำไปประยุกต์ใช้หลากหลาย ยกตัวอย่างเช่น การแปลงเสียงออกมาเป็นข้อความ, การสร้างโปรแกรมจำแนกเสียงอัตโนมัติ สำหรับผู้พิการทางการได้ยิน, การยืนยันตัวตนด้วยเสียง หรือ การนำเสียงของผู้ร้ายไปหาผู้ต้องสงสัยในคดีอาชญากรรม เป็นต้น ซึ่งสิ่งเหล่านี้ ต่างต้องการอัลกอริทึมที่สามารถจำแนกเสียงออกเป็นกลุ่มต่างๆ ได้อย่างรวดเร็วและแม่นยำด้วยกันทั้งสิ้น ดังนั้น พวกเราจึงสนใจที่จะทดลองวิธีใหม่ๆ เพื่อที่จะเพิ่มประสิทธิภาพและความแม่นยำในการจำแนกเสียง

ตัวแปรในการทดลอง

ตัวแปรต้น : ชนิดของ Window function, ชนิดของ Distance function, ค่าคงที่ k สำหรับ k -NN และ อัตราส่วนของ Noise ในชุดของเสียงทดสอบ

ตัวแปรตาม : ความแม่นยำในการจำแนกเสียง (จำนวนของเสียงตัวอย่าง ที่สามารถตอบได้ถูกต้อง)

ตัวแปรควบคุม : ชุดของเสียงตั้งต้น ที่นำมาใช้ในการทดลอง

สมมุติฐาน

ความแม่นยำในการจำแนกเสียงเครื่องดนตรี จะมีค่าเพิ่มมากขึ้น เมื่อใช้ window function ร่วมในการประมวลผล ทั้งในสภาพที่มีเสียงรบกวนเยอะ และในสภาพปกติ

ซอฟต์แวร์/อุปกรณ์ที่ใช้ในการทดลอง

1. คอมพิวเตอร์ พร้อมไมโครโฟนในการอัดเสียง
2. Compiler ภาษา C++ (MinGW)
3. โปรแกรม Node.js
4. โปรแกรม Audacity
5. ไลบรารี libsndfile
6. เครื่องดนตรี 3 ชนิด ได้แก่ กีตาร์ เปียโน และ ไวโอลิน

ขั้นตอนการดำเนินงาน

สำหรับการดำเนินงาน สามารถแบ่งออกเป็น 5 ขั้นตอนหลักๆด้วยกัน ได้แก่

- ขั้นที่ 1 ศึกษางานวิจัยและทฤษฎีที่เกี่ยวข้อง และออกแบบแนวทางในการพัฒนาโปรแกรมทดสอบ
- ขั้นที่ 2 ทำการเก็บรวบรวมข้อมูลเสียงที่จำเป็นสำหรับการประมวลผล (Dataset)
- ขั้นที่ 3 พัฒนาโปรแกรมคอมพิวเตอร์ เพื่อใช้ในการทำการทดลอง
- ขั้นที่ 4 ทดสอบโปรแกรมโดยใช้ข้อมูลเสียงตัวอย่าง
- ขั้นที่ 5 สรุปและอภิปรายผล

มีรายละเอียดดังนี้

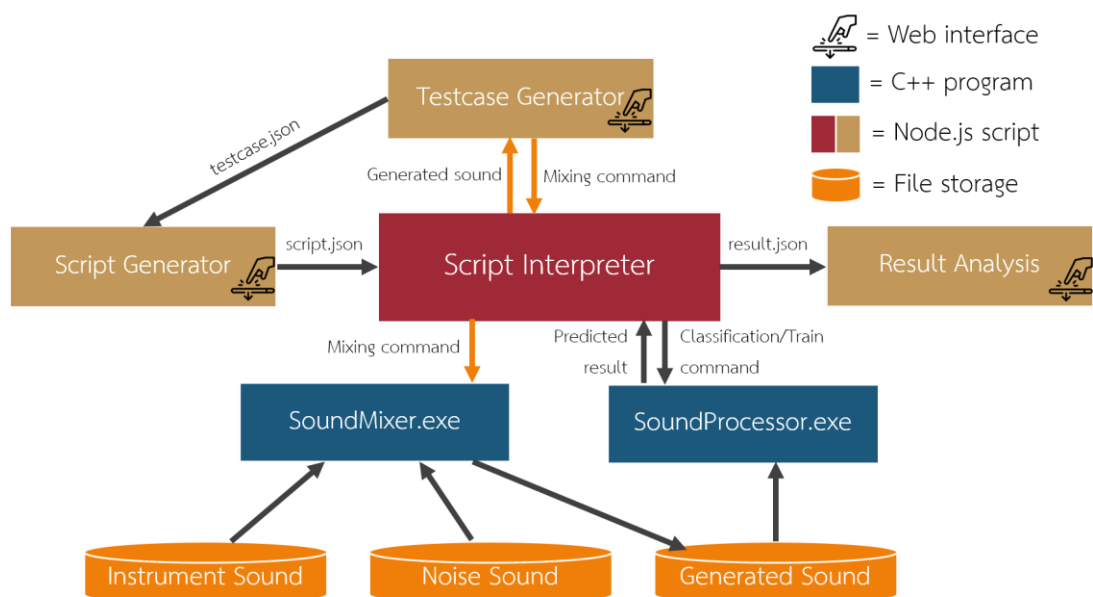
1) ศึกษางานวิจัยและทฤษฎีที่เกี่ยวข้องและออกแบบแนวทางในการพัฒนาโปรแกรมทดสอบ

เพื่อให้เกิดขั้นตอนวิธีที่มีประสิทธิภาพในการจำแนกชนิดของเสียง จำเป็นต้องมีการศึกษาหา งานวิจัยที่เกี่ยวข้องก่อน โดยมีหัวข้อหลักๆดังนี้

1.1) ศึกษาทฤษฎีในการประมวลผลสัญญาณและจำแนกข้อมูล ได้แก่ Fast Fourier Transform (FFT), k-Nearest Neighbors (k-NN) และ Windows functions ต่างๆ

1.2) ศึกษาการพัฒนาโปรแกรมในภาษา C++ และ Nodejs ให้สามารถประมวลผลและ วิเคราะห์เสียงได้ รวมไปถึงการใช้ library ต่างๆ เช่น libsndfile, chart.js, express เป็นต้น

และในส่วนของการออกแบบโปรแกรม มีแผนผังโครงสร้างของโปรแกรมดังรูป



โดยโปรแกรม จะแบ่งออกเป็นสองส่วนหลักๆ

ได้แก่ โปรแกรมภาษา C++ (สี่เหลี่ยมสีฟ้า) และ โปรแกรมที่เขียนใน Node.js (สี่เหลี่ยมที่เหลือง) ซึ่งโปรแกรมภาษา C++ จะทำหน้าที่ที่เกี่ยวข้องกับการประมวลผลเสียงและการจำแนกเสียง (SoundProcessor.exe) รวมถึงผสมเสียงเครื่องดนตรีเข้ากับเสียงรบกวน(Noise) เพื่อใช้ในการทดลองอีกด้วย (SoundMixer.exe)

ในส่วนของ Node.js ประกอบด้วย 4 ส่วนย่อยๆ ดังนี้

Script Interpreter: ทำหน้าที่ประสานการทำงานระหว่างโปรแกรม Node.js และ C++

Testcase Generator: ส่งคำสั่งไปยัง Sound Mixer (ผ่านทาง Interpreter) เพื่อผสมเสียงเครื่องดนตรี กับ Noise เข้าด้วยกัน จนเกิดเป็นชุดของเสียง (Test case) ขึ้น

Script Generator: นำชุดของเสียงที่สร้างจาก Testcase Generator มาสร้างเป็นชุดคำสั่ง เพื่อที่จะใช้ทดลองตามตัวแปรและสมมุติฐานที่ได้วางไว้

Result Analysis: รับผลลัพธ์ที่ได้จากการทดลองมาวิเคราะห์หาความถูกต้องและส่งข้อมูลกลับไปทางผู้ใช้ผ่านทาง web interface

2) ขั้นเก็บรวบรวมข้อมูลเสียงที่เกี่ยวข้องกับการประมวลผล

รวบรวมเสียงตัวอย่างจากเครื่องดนตรีต่างๆ เพื่อใช้ทดสอบความแม่นยำและประสิทธิภาพของโปรแกรม โดยรวบรวมเสียงจากเครื่องดนตรี 3 ชนิด ได้แก่ กีตาร์ เปียโน และ ไวโอลิน ชนิดละ 50 เสียง โดยในแต่ละเสียง จะเก็บเป็นไฟล์ .wav ที่มีเพียงระดับเสียงเดียวและเครื่องดนตรีชนิดเดียวต่อหนึ่งไฟล์เสียง รวมเป็นทั้งสิ้น 150 ไฟล์ ซึ่งเสียงที่รวบรวมมาจะไม่มีเสียงรบกวน และ มีความถี่(ตัวโน้ต) แบบสุ่ม รวมถึงหาเสียงรบกวน (Noise) อีกจำนวนทั้งสิ้น 10 เสียง เพื่อนำมาผสมกับเสียง

3) ขั้นพัฒนาโปรแกรมคอมพิวเตอร์

พัฒนาโปรแกรมคอมพิวเตอร์ เพื่อนำไปประมวลผลและแยกเสียง โดยมีขั้นตอนย่อยๆดังนี้

3.1) พัฒนาโปรแกรมในส่วนของการประมวลผลสัญญาณเสียงด้วยภาษา C++ โดยใช้ libsndfile ในการอ่านและเขียนไฟล์เสียง รวมถึงเขียนส่วนการประมวลผลด้วย Fast Fourier Transform และจำแนกเสียงด้วย K-NN

3.2) พัฒนาโปรแกรมส่วนของหน้าเว็บและส่วนทดสอบอัลกอริทึม โดยใช้ Node.js ตามแผนผังที่วางไว้ในขั้นตอนที่ 2

3.3) ทดลองโปรแกรมในแต่ละส่วน โดยใช้ test case ที่มีขนาดไม่ใหญ่มากนัก เพื่อที่จะแก้ไขจุดบกพร่องในโปรแกรม

4) ขั้นทดสอบโปรแกรม

ทดสอบความถูกต้องของโปรแกรม โดยจะมี Test Data ทั้งหมด 5 ชุด โดยใน 1 ชุด จะแบ่งเป็นกลุ่มเสียงสองกลุ่ม ขนาดกลุ่มละ 75 เสียง แบ่งเป็นเครื่องดนตรี 3 ชนิด ชนิดละ 25 เสียง ซึ่งกลุ่มแรกจะเป็นข้อมูลที่ใช้ไปในโปรแกรม เพื่อฝึกโปรแกรมให้จำแนกเสียง (Train) และกลุ่มสองจะเป็นข้อมูลที่ใช้ทดสอบความถูกต้องของโปรแกรม (Test)

ส่วน Test case ทั้งหมด 5 ชุด มีความแตกต่างกันคือ

1. เสียงใน Train และ Test เป็นเสียงต้นฉบับที่ไม่มีเสียงรบกวนใดๆ
2. เสียงใน Train เป็นเสียงต้นฉบับและเสียงใน Test เป็นเสียงที่มี Noise 10%
3. เสียงใน Train เป็นเสียงต้นฉบับและเสียงใน Test เป็นเสียงที่มี Noise 25%
4. เสียงใน Train เป็นเสียงต้นฉบับและเสียงใน Test เป็นเสียงที่มี Noise 50%
5. เสียงใน Train เป็นเสียงต้นฉบับและเสียงใน Test เป็นเสียงที่มี Noise 75%

โดยใน Test case 1 ชุด เราจะทำการทดลองทั้งหมด 4 รอบ ในแต่ละรอบจะทำการสุ่มชุดของเสียงที่เป็น test และ train ขึ้นมาใหม่ แต่จะยังคงอัตราส่วนเครื่องดนตรีและเงื่อนไขของแต่ละ Test case ไว้

และในการทดลอง 1 รอบ จะมีการทดสอบย่อยๆอีก 60 ครั้ง โดยในแต่ละครั้ง จะทดลองชุดของตัวแปรต้นที่แตกต่างกันไป ซึ่งมีตัวแปรต้นดังนี้

Window function: Rectangular window, Triangular window, Hanning window, Hamming window, Blackman-Harris window, Flat top window

Distance function: Euclidean Distance, Manhattan Distance

ค่าคงที่ k : 1, 2, 3, 4, 5

ในการทดสอบย่อยๆแต่ละครั้งนั้น จะมี Train โปรแกรมด้วยชุดเสียงทั้งหมด 75 เสียงก่อน และจากนั้น จึงตามด้วยการ Test โปรแกรมด้วยเสียงอีก 75 เสียง (ตามข้อมูลเสียงใน Testcase)

5) ขั้นสรุปและอภิปรายผล

ในขั้นตอนนี้จะทำการวิเคราะห์ความถูกต้องของตัวแปรต้นแต่ละชุด โดยใช้เครื่องมือในการวิเคราะห์ได้แก่ การสร้าง Confusion Matrix, การหาร้อยละความถูกต้องและร้อยละของกรณีที่ไม่สามารถทำนายได้ (ใช้ KNN แล้วเกิดฐานนิยามหลายตัว) ดังสมการ

$$\text{ร้อยละความถูกต้อง} = \frac{\text{จำนวนเสียงที่ตอบถูก}}{\text{จำนวนเสียงที่ใช้ทดสอบ}} \times 100$$

$$\text{ร้อยละกรณีที่ไม่สามารถทำนายได้} = \frac{\text{จำนวนกรณีที่ } KNN \text{ พบหลายฐานนิยาม}}{\text{จำนวนเสียงที่ใช้ทดสอบ}} \times 100$$

หลังจากนั้นนำค่าเหล่านี้มาวิเคราะห์ อภิปราย และสรุปผลต่อไป

ขั้นตอนในการทำงานของอัลกอริทึมจำแนกเสียง

1. รับเสียง, Window function, Distance function และ K constant เข้ามา ผ่านทาง command-line arguments
2. ตัดเสียงออกมาช่วงสั้นๆ จากจุดกึ่งกลางของเสียงที่รับเข้ามา
3. Normalize สัญญาณเสียงที่รับเข้ามาให้อยู่ในช่วง $[-1,1]$
4. ใช้ Window function เข้ามาคูณกับข้อมูลเสียง
5. นำเสียงที่คูณด้วย Window function แล้วมาประมวลผลด้วย Fast Fourier Transform (FFT) เพื่อแปลงคลื่นเสียงที่รับเข้ามา จากที่มีลักษณะ Time-domain เป็นลักษณะ Frequency-domain แทน
6. ตัดข้อมูลที่ได้จากการประมวลผลให้เหลือเพียงช่วงความถี่ที่มนุษย์ได้ยิน คือ 20–20,000 Hz
7. กรณีที่เป็นการ Train ข้ามไปยังขั้นตอนที่ 10 และหากเป็นการ Test ให้ข้ามไปยังขั้นตอนที่ 8
8. นำข้อมูลมาจำแนกชนิดข้อมูล โดยเทียบกับข้อมูลที่เก็บไว้ในฐานข้อมูล เพื่อระบุชนิดของเครื่องดนตรีของเสียงที่รับเข้ามา โดยนำ k ตัวแรก ที่มีความใกล้เคียงกับเสียงที่รับเข้ามามากที่สุด (ประเมินโดยใช้ Distance function) มาหาฐานนิยม หรือพูดอีกนัยหนึ่ง คือการหา k-Nearest Neighbors นั่นเอง
9. ส่งชนิดของที่ทำนายได้กลับไปทาง Standard Output
10. นำข้อมูลความถี่ของเสียง บันทึกลงไปในฐานข้อมูล

ผลการทดลอง

เมื่อทำการทดลองครบตามขั้นตอนแล้ว สามารถสรุปผลได้ค่าดังนี้

ร้อยละของความแม่นยำใน Test case ที่ 1 (ไม่มีเสียงรบกวน)

เมื่อใช้ Euclidean Distance ในการจำแนกเสียง

ค่า k Window	1	2	3	4	5
Rectangular	81.33	50.67	58.67	42.33	48.00
Triangular	80.33	48.33	63.00	42.00	51.00
Hanning	80.67	43.33	58.00	36.67	44.00
Hamming	79.33	44.33	59.33	38.00	45.00
Blackman-Harris	81.33	43.00	56.00	35.67	47.00
Flat top	96.00	79.33	84.67	73.00	79.00

ร้อยละของความแม่นยำใน Test case ที่ 1 (ไม่มีเสียงรบกวน)

เมื่อใช้ Manhattan Distance ในการจำแนกเสียง

ค่า k Window	1	2	3	4	5
Rectangular	84.00	57.67	62.33	51.33	55.33
Triangular	86.00	56.67	62.67	49.33	48.67
Hanning	86.67	57.00	64.00	50.33	48.67
Hamming	86.00	54.67	61.33	48.00	46.00
Blackman-Harris	87.33	58.33	66.33	52.33	52.33
Flat top	83.33	62.33	69.00	54.67	56.67

ร้อยละของความแม่นยำใน Test case ที่ 2 (มีเสียงรบกวน 10%)

เมื่อใช้ Euclidean Distance ในการจำแนกเสียง

ค่า k Window	1	2	3	4	5
Rectangular	78.67	46.00	54.33	36.33	41.67
Triangular	81.00	45.00	59.33	37.67	46.00
Hanning	78.00	41.67	53.00	33.67	40.67
Hamming	78.67	40.67	52.33	33.33	41.33
Blackman-Harris	80.33	42.33	57.00	32.67	42.67
Flat top	90.33	73.67	77.67	66.00	69.67

ร้อยละของความแม่นยำใน Test case ที่ 2 (มีเสียงรบกวน 10%)

เมื่อใช้ Manhattan Distance ในการจำแนกเสียง

ค่า k Window	1	2	3	4	5
Rectangular	79.00	51.00	61.67	53.00	56.00
Triangular	80.67	53.33	64.67	51.00	55.00
Hanning	80.00	53.67	63.00	50.33	57.00
Hamming	80.00	57.33	64.33	54.67	59.00
Blackman-Harris	81.67	52.67	64.67	47.67	55.33
Flat top	84.33	66.33	75.00	62.00	68.00

ร้อยละของความแม่นยำใน Test case ที่ 3 (มีเสียงรบกวน 25%)

เมื่อใช้ Euclidean Distance ในการจำแนกเสียง

ค่า k Window	1	2	3	4	5
Rectangular	70.67	40.33	46.33	34.67	37.33
Triangular	78.00	37.00	54.33	33.67	38.00
Hanning	73.67	32.33	46.67	25.67	35.00
Hamming	73.67	33.33	47.67	26.00	34.00
Blackman-Harris	77.33	35.00	51.33	32.00	39.00
Flat top	83.33	61.00	75.33	60.00	66.00

ร้อยละของความแม่นยำใน Test case ที่ 3 (มีเสียงรบกวน 25%)

เมื่อใช้ Manhattan Distance ในการจำแนกเสียง

ค่า k Window	1	2	3	4	5
Rectangular	62.67	37.33	50.67	41.67	46.33
Triangular	62.67	31.33	48.00	32.67	36.00
Hanning	62.33	31.00	45.00	28.67	35.67
Hamming	62.33	32.67	45.33	32.33	38.00
Blackman-Harris	63.33	30.67	45.67	26.33	34.67
Flat top	67.33	54.00	61.33	56.67	58.33

ร้อยละของความแม่นยำใน Test case ที่ 4 (มีเสียงรบกวน 50%)

เมื่อใช้ Euclidean Distance ในการจำแนกเสียง

ค่า k Window	1	2	3	4	5
Rectangular	57.00	36.00	36.67	33.67	34.33
Triangular	54.67	24.67	39.00	26.00	34.00
Hanning	53.33	23.33	38.00	26.00	33.00
Hamming	54.00	23.33	36.67	26.00	32.33
Blackman-Harris	54.33	26.33	36.33	23.33	34.33
Flat top	55.33	36.00	44.00	38.33	42.00

ร้อยละของความแม่นยำใน Test case ที่ 4 (มีเสียงรบกวน 50%)

เมื่อใช้ Manhattan Distance ในการจำแนกเสียง

ค่า k Window	1	2	3	4	5
Rectangular	56.00	30.33	39.00	31.67	34.67
Triangular	51.00	23.33	34.33	19.33	28.33
Hanning	50.67	22.00	32.67	21.33	27.67
Hamming	51.67	25.00	34.67	22.00	28.33
Blackman-Harris	47.00	20.33	32.67	18.33	27.00
Flat top	47.67	41.33	53.00	48.00	54.00

ร้อยละของความแม่นยำใน Test case ที่ 5 (มีเสียงรบกวน 75%)

เมื่อใช้ Euclidean Distance ในการจำแนกเสียง

ค่า k Window	1	2	3	4	5
Rectangular	36.00	33.33	33.33	33.33	33.33
Triangular	38.67	18.67	42.00	25.33	41.00
Hanning	42.67	22.67	40.33	25.67	36.00
Hamming	42.67	23.67	41.67	26.33	35.00
Blackman-Harris	38.33	19.67	40.67	21.33	37.67
Flat top	41.67	20.33	32.33	26.33	31.00

ร้อยละของความแม่นยำใน Test case ที่ 5 (มีเสียงรบกวน 75%)

เมื่อใช้ Manhattan Distance ในการจำแนกเสียง

ค่า k Window	1	2	3	4	5
Rectangular	35.00	21.00	36.33	27.33	33.00
Triangular	36.67	19.33	30.67	20.00	30.00
Hanning	34.00	18.00	30.67	18.67	29.33
Hamming	33.33	18.67	29.33	17.67	30.00
Blackman-Harris	35.00	15.67	30.33	23.00	31.00
Flat top	36.33	27.00	39.67	32.33	41.67

ร้อยละของกรณีที่ไม่สามารถทำนายได้ใน Test case ที่ 1 (ไม่มีเสียงรบกวน)

เมื่อใช้ Euclidean Distance ในการจำแนกเสียง

ค่า k Window	1	2	3	4	5
Rectangular	0.00	34.33	0.00	21.00	0.67
Triangular	0.00	43.33	1.33	33.00	12.33
Hanning	0.00	46.00	4.67	34.33	4.67
Hamming	0.00	45.33	4.67	33.67	4.67
Blackman-Harris	0.00	49.33	2.67	38.67	10.00
Flat top	0.00	19.33	2.00	20.00	4.67

ร้อยละของกรณีที่ไม่สามารถทำนายได้ใน Test case ที่ 1 (ไม่มีเสียงรบกวน)

เมื่อใช้ Manhattan Distance ในการจำแนกเสียง

ค่า k Window	1	2	3	4	5
Rectangular	0.00	32.33	6.67	18.67	11.00
Triangular	0.00	36.00	5.33	21.00	15.00
Hanning	0.00	36.00	4.33	21.67	12.00
Hamming	0.00	38.33	6.67	21.33	12.33
Blackman-Harris	0.00	35.00	4.67	23.33	14.00
Flat top	0.00	31.67	2.67	21.67	6.67

ร้อยละของกรณีที่ไม่สามารถทำนายได้ใน Test case ที่ 2 (มีเสียงรบกวน 10%)

เมื่อใช้ Euclidean Distance ในการจำแนกเสียง

ค่า k Window	1	2	3	4	5
Rectangular	0.00	36.00	0.00	22.67	0.00
Triangular	0.00	44.00	4.33	32.67	12.33
Hanning	0.00	43.67	2.00	32.33	9.00
Hamming	0.00	45.00	2.00	34.33	7.33
Blackman-Harris	0.00	45.33	4.67	35.67	12.67
Flat top	0.00	20.00	4.00	18.33	6.00

ร้อยละของกรณีที่ไม่สามารถทำนายได้ใน Test case ที่ 2 (มีเสียงรบกวน 10%)

เมื่อใช้ Manhattan Distance ในการจำแนกเสียง

ค่า k Window	1	2	3	4	5
Rectangular	0.00	36.00	3.33	15.67	4.00
Triangular	0.00	32.67	2.33	22.00	7.00
Hanning	0.00	29.67	2.00	21.33	5.67
Hamming	0.00	27.33	2.33	18.33	5.67
Blackman-Harris	0.00	32.67	2.33	26.33	5.00
Flat top	0.00	20.00	1.33	18.33	4.67

ร้อยละของกรณีที่ไม่สามารถทำนายได้ใน Test case ที่ 3 (มีเสียงรบกวน 25%)

เมื่อใช้ Euclidean Distance ในการจำแนกเสียง

ค่า k Window	1	2	3	4	5
Rectangular	0.00	37.67	0.00	16.33	0.00
Triangular	0.00	52.00	6.67	34.67	17.67
Hanning	0.00	53.00	4.67	36.00	10.00
Hamming	0.00	51.67	4.67	34.00	10.33
Blackman-Harris	0.00	54.00	8.33	34.33	15.67
Flat top	0.00	30.00	1.33	23.00	6.00

ร้อยละของกรณีที่ไม่สามารถทำนายได้ใน Test case ที่ 3 (มีเสียงรบกวน 25%)

เมื่อใช้ Manhattan Distance ในการจำแนกเสียง

ค่า k Window	1	2	3	4	5
Rectangular	0.00	40.67	3.00	15.67	6.67
Triangular	0.00	46.00	4.00	25.00	7.67
Hanning	0.00	44.00	2.67	24.00	5.67
Hamming	0.00	43.33	2.67	23.67	6.33
Blackman-Harris	0.00	46.67	2.67	30.00	6.67
Flat top	0.00	20.67	0.67	9.67	2.00

ร้อยละของกรณีที่ไม่สามารถทำนายได้ใน Test case ที่ 4 (มีเสียงรบกวน 50%)

เมื่อใช้ Euclidean Distance ในการจำแนกเสียง

ค่า k Window	1	2	3	4	5
Rectangular	0.00	22.67	0.00	5.33	0.00
Triangular	0.00	53.00	5.67	33.33	10.00
Hanning	0.00	57.33	5.33	28.67	2.67
Hamming	0.00	57.67	7.00	30.00	3.33
Blackman-Harris	0.00	48.67	2.67	31.00	4.67
Flat top	0.00	32.00	4.00	17.33	6.33

ร้อยละของกรณีที่ไม่สามารถทำนายได้ใน Test case ที่ 4 (มีเสียงรบกวน 50%)

เมื่อใช้ Manhattan Distance ในการจำแนกเสียง

ค่า k Window	1	2	3	4	5
Rectangular	0.00	37.67	3.00	15.33	4.00
Triangular	0.00	50.67	3.33	37.33	8.00
Hanning	0.00	48.00	2.33	33.00	8.67
Hamming	0.00	46.00	2.67	29.00	7.67
Blackman-Harris	0.00	48.00	1.33	35.00	6.67
Flat top	0.00	17.67	0.33	10.67	0.67

ร้อยละของกรณีที่ไม่สามารถทำนายได้ใน Test case ที่ 5 (มีเสียงรบกวน 75%)

เมื่อใช้ Euclidean Distance ในการจำแนกเสียง

ค่า k Window	1	2	3	4	5
Rectangular	0.00	6.67	0.00	0.00	0.00
Triangular	0.00	56.67	3.33	45.00	6.00
Hanning	0.00	46.67	0.00	35.00	0.33
Hamming	0.00	48.67	0.33	35.33	0.33
Blackman-Harris	0.00	53.00	3.67	48.00	7.00
Flat top	0.00	47.33	10.00	24.00	9.67

ร้อยละของกรณีที่ไม่สามารถทำนายได้ใน Test case ที่ 5 (มีเสียงรบกวน 75%)

เมื่อใช้ Manhattan Distance ในการจำแนกเสียง

ค่า k Window	1	2	3	4	5
Rectangular	0.00	39.67	1.33	19.67	2.00
Triangular	0.00	49.33	0.00	32.33	1.67
Hanning	0.00	51.00	0.33	36.67	4.67
Hamming	0.00	47.67	0.33	38.33	2.67
Blackman-Harris	0.00	49.33	0.00	26.67	2.33
Flat top	0.00	30.33	0.00	20.33	0.00

วิเคราะห์ผลการทดลอง

จากผลการทดลอง จะพบว่ามีข้อสังเกตที่น่าสนใจหลายประการดังนี้

1. ในกรณีของชุดทดสอบที่ 1 (ไม่มีเสียงรบกวน) เมื่อใช้ Euclidean Distance และ Flat top windowing จะมีอัตราการตอบถูกที่สูงที่สุด เมื่อเทียบกับ Windows และ Distance Function อื่น โดยให้ค่าสูงสุดเมื่อ $k = 1$ คือถึงร้อยละ 96.00
2. โดยเฉลี่ย Flat top windowing จะให้ความแม่นยำสูงกว่า Window function อื่นๆ ในทุกเงื่อนไขและสภาพแวดล้อม แต่สำหรับ window function อีก 5 ตัวที่เหลือนั้น จะมีความแม่นยำใกล้เคียงกับกรณีที่ไม่ได้ใช้ window function มาก
3. เมื่อ $k=2$ และ $k=4$ จะมีร้อยละความแม่นยำ ต่ำกว่าในกรณี $k=1$, $k=3$ และ $k=5$ ซึ่งหากพิจารณาจาก ร้อยละของกรณีที่ไม่สามารถทำนายได้ (Error rate) แล้ว จะพบว่าสาเหตุหลักเกิดจากที่เมื่อ $k=2$ และ $k=4$ จะมี Error rate สูงกว่ามากนั่นเอง
4. ค่าเฉลี่ยของร้อยละความแม่นยำในแต่ละ Test case เป็นดังนี้

1: ไม่มี Noise	2: Noise 10%	3: Noise 25%	4: Noise 50%	5: Noise 75%
60.56	58.90	47.33	36.13	30.70

ซึ่งจะเห็นได้ว่า ค่าของความถูกต้องจะแปรผกผันกับอัตราส่วนของ Noise

5. เมื่ออัตราส่วนของเสียงรบกวนมีค่าเป็น 75% ค่าเฉลี่ยของความถูกต้องของอัลกอริทึมในทุกๆ คู่ของตัวแปรต้น จะมีค่าใกล้เคียงกับ 33.33% ซึ่งเท่ากับความน่าจะเป็นเมื่อเดาสุ่ม จึงสามารถสรุปว่าการใช้อัลกอริทึมนี้ จะไม่สามารถจำแนกเสียงได้อย่างมีประสิทธิภาพ ในสภาพแวดล้อมที่มีเสียงรบกวนสูง
6. หากลองสร้าง confusion matrix โดยนำผลลัพธ์ของการทดลองทั้งหมดมารวมกัน ได้ดังนี้

Predict \ Expect	Piano	Guitar	Violin
Piano	21.34 %	1.34 %	6.13 %
Guitar	10.25 %	14.37 %	4.01 %
Violin	15.72 %	0.62 %	11.02 %

ไม่สามารถทำนายได้ : 15.20 %

ซึ่งจะพบว่ากรณีที่ตอบผิดส่วนใหญ่ เกิดขึ้นจาก 2 กรณีหลักๆด้วยกัน คือกรณีที่ไม่สามารถทำนายได้ (15.20 %) และ กรณีที่อัลกอริทึมทำนายได้ว่าเป็นเสียงเปียโน แต่เสียงแท้จริงเป็นคนละประเภท (25.97 %)

อภิปรายผลการทดลองและข้อเสนอแนะ

การทดลองนี้นั้น ถูกทดลองขึ้น โดยมีขอบเขตของการทดลองค่อนข้างจำกัด คือเป็นการจำแนกเสียงของเครื่องดนตรีเพียง 3 ชนิด และจำแนกเสียงที่มีลักษณะเป็นโน้ตเดี่ยวต่อ 1 ไฟล์เท่านั้น ดังนั้นในอนาคต จึงสามารถต่อยอดไปยังในกรณีที่เสียงที่นำมาจำแนกมีหลายชนิด และไม่จำกัดประเภทของเสียงที่จำแนกได้ รวมถึงอาจจะเพิ่มจำนวนของ test data ขึ้นได้มากกว่านี้ ซึ่งจะเพิ่มความถูกต้องของอัลกอริทึมได้ และนอกจากนี้ เนื่องจากเมื่อ Noise เพิ่มขึ้น จะทำให้อัตราการตอบถูกลดลงอย่างเห็นได้ชัด การเพิ่มส่วนของอัลกอริทึมที่ใช้ในการลด Noise จึงเป็นสิ่งที่ควรกระทำ และจากที่กรณีที่ไม่สามารถทำนายได้ เป็นหนึ่งในตัวแปรสำคัญต่อความแม่นยำในการทำนาย ดังนั้นจึงควรมีการจัดการกับกรณีเหล่านี้ด้วย

สรุปผลการทดลอง

จากการวิเคราะห์และอภิปรายผลการทดลอง พบว่าการทดลองนั้นเป็นไปตามสมมุติฐานที่ตั้งไว้นั้นคือการใช้ Window function ช่วยเพิ่มความแม่นยำของการจำแนกเสียงได้จริง แต่อย่างไรก็ตาม window function ที่มีประสิทธิภาพในการเพิ่มความแม่นยำอย่างเห็นได้ชัด มีเพียง Flat top window เท่านั้น ดังนั้นสำหรับการจำแนกเสียงของเครื่องดนตรี 3 ชนิด พบว่า ควรใช้ $k=1$, Euclidean Distance และ Flat top windowing ซึ่งจะมีอัตราการตอบถูกที่สูงที่สุดในสภาพแวดล้อมที่ไม่มีเสียงรบกวนถึงร้อยละ 96.00

กิตติกรรมประกาศ

โครงการเรื่องการศึกษาผลของ Window function ต่อความแม่นยำในการจำแนกเสียงเครื่องดนตรีด้วย FFT และ K-NN Classifier สามารถดำเนินการจนประสบความสำเร็จได้ ด้วยความช่วยเหลือของครูที่ปรึกษาคือ ครูอภินันท์ เงินมูล และครูที่ปรึกษาพิเศษอีกสองท่าน ได้แก่ ครูธัญญา ศรีหมากสุข และครูชนิตาภา กัญจนวัตตะ ที่คอยให้คำปรึกษาและคำแนะนำในการทำโครงการนี้ ทางคณะผู้ศึกษาได้ขอขอบคุณเป็นอย่างสูงในความช่วยเหลือครั้งนี้ขอขอบคุณครูชัยชนะ นุชฉัยยา ที่กรุณาให้ความช่วยเหลือในการตรวจสอบแก้ไขโครงการนี้ให้ถูกต้องแล้วเสร็จสมบูรณ์

ขอขอบคุณทางโรงเรียนสามเสนวิทยาลัยที่เอื้อเฟื้อสถานที่ในการศึกษาโครงการนี้

ขอขอบคุณโครงการ YSC และ NECTEC ที่ให้ทุนสนับสนุนในการทำโครงการ

คณะผู้ศึกษาขอขอบคุณทุกท่านที่มีส่วนร่วมในการศึกษาโครงการนี้จนสำเร็จลุล่วง

คณะผู้จัดทำ

นายจิตรภาณุ อัสวพิชญโชติ

นายเขมรัฐ บุณยะผลึก

นายพุฒิพัฒน์ สิริเศรษฐภักดี

บรรณานุกรม

Adam Głowacz, Witold Głowacz, Andrzej Głowacz. (2553). Sound Recognition of Musical Instruments with Application of FFT and K-NN Classifier with Cosine Distance.

Automatyka.

Erik de Castro Lopo. (2545). Introduction to Audio Digital Signal Processing on Linux. (ออนไลน์).

แหล่งที่มา : <http://www.mega-nerd.com/Res/IADSPL/>

Robert A. Schilling, Sandra L. Harris. (2556). Introduction to Digital Signal Processing using MATLAB 2nd Edition. พิมพ์ครั้งที่ 2. Cengage Learning. Boston. US.

Toni Heittola, Anssi Klapuri, Tuomas Virtanen. (2552). Musical Instrument Recognition in Polyphonic Audio Using Source-Filter Model for Sound Separation. Kobe

International

Conference Center. Kobe. Japan.

Wikipedia. (2559). Window function. (ออนไลน์). แหล่งที่มา:

https://en.wikipedia.org/wiki/Window_function