Τι είναι οι Mel Frequencies Cepstral Coefficients (MFCCs); Πως χρησιμοποιούνται στην επεξεργασία ήχου; Γιατί χρησιμοποιούνται στην αναγνώριση συναισθημάτων με συνελικτικά νευρωνικά δίκτυα;

Μπορείς να μου εξηγήσεις επίσης στην ενότητα 3 "Methodology" τις 2 εικόνες figure 1 και figure 2

ΑΝΑΦΕΡΕΤΕ ΣΤΟ electronics-12-04376-with-cover PAPER

Ας απαντήσουμε στα ερωτήματά σου βήμα-βήμα:

**🔹 Τι είναι τα Mel Frequency Cepstral Coefficients (MFCCs);**

Τα **MFCCs** είναι **συντελεστές που περιγράφουν το φάσμα του ήχου** με βάση την ανθρώπινη ακουστική αντίληψη. Συγκεκριμένα:

* Το φάσμα του ήχου μετασχηματίζεται ώστε να αντικατοπτρίζει το πώς το ανθρώπινο αυτί αντιλαμβάνεται τις **συχνότητες** (με τη μη-γραμμική **Mel κλίμακα**).
* Ο ήχος περνά από **Fourier Transform**, **Mel filterbank**, **λογαριθμισμό** και τέλος **Discrete Cosine Transform (DCT)**, εξάγοντας τους MFCCs.
* Συνήθως χρησιμοποιούνται οι πρώτοι ~12–40 συντελεστές.

**🔹 Πώς χρησιμοποιούνται στην επεξεργασία ήχου;**

Οι MFCCs λειτουργούν ως **χαρακτηριστικά εισόδου** για αλγορίθμους μηχανικής μάθησης. Είναι εξαιρετικά αποτελεσματικοί για:

* Αναγνώριση ομιλίας (Speech Recognition)
* Αναγνώριση ηχείου (Speaker Recognition)
* Αναγνώριση συναισθήματος (Emotion Recognition)

Διότι αποτυπώνουν **χαμηλού επιπέδου χαρακτηριστικά** φωνής, όπως η τονικότητα, ένταση και φασματική δομή.

**🔹 Γιατί χρησιμοποιούνται στην αναγνώριση συναισθημάτων με συνελικτικά νευρωνικά δίκτυα (CNNs);**

Τα **CNNs** είναι πολύ αποδοτικά στην αναγνώριση **χωρικών μοτίβων**, όπως σε εικόνες. Όταν οι MFCCs αναπαρίστανται ως **εικόνα (π.χ. MFCCs vs Time)**, τα CNN μπορούν:

* Να **ανιχνεύσουν χρονικά μοτίβα** στις μεταβολές των φωνητικών χαρακτηριστικών.
* Να **εξάγουν υψηλότερου επιπέδου αναπαραστάσεις**, χρήσιμες για ταξινόμηση συναισθημάτων.

Η προσθήκη μηχανισμών **προσοχής (attention)** επιτρέπει στο δίκτυο να "εστιάζει" στις πιο σημαντικές χρονικές στιγμές του ήχου.

**🔹 Εξήγηση των Εικόνων στην Ενότητα 3 "Methodology"**

**Figure 1** – *"Audio file from the RAVDESS"*

* Παρουσιάζει το **ηχητικό σήμα** ενός αρχείου από τη βάση δεδομένων **RAVDESS** σε μορφή κυματομορφής.
* Άξονες: Χρόνος (x) και Αμplitude (y)
* Δείχνει **πώς μεταβάλλεται η ένταση του ήχου με τον χρόνο**.

**Figure 2** – *"MFCCs of the RAVDESS audio file in Figure 1"*

* Αναπαριστά τους **πρώτους 40 MFCCs** (χαρακτηριστικά του ήχου) ως **χρονική ακολουθία**.
* Άξονες: Χρόνος (x) και MFCC index (y)
* Χρωματική κωδικοποίηση: Δείχνει την **ένταση κάθε MFCC** στο χρόνο.
* Ουσιαστικά, πρόκειται για μια **εικόνα-χάρτης χαρακτηριστικών**, την οποία επεξεργάζονται τα CNN μοντέλα.

Αν θέλεις, μπορώ να αναλύσω περαιτέρω την αρχιτεκτονική των δικτύων ή το γιατί η προσθήκη μηχανισμού προσοχής βελτιώνει τα αποτελέσματα. Θες να συνεχίσουμε με αυτό;

Θέλω να μου εξηγήσεις αναλυτικά πρώτα πως παράγονται οι Mel Frequencies

Cepstral Coefficients (MFCCs)

Βεβαίως! Ας δούμε **αναλυτικά και βήμα-βήμα** πώς παράγονται οι **Mel Frequency Cepstral Coefficients (MFCCs)** από ένα αρχείο ήχου.

**🎯 Τι είναι οι MFCCs;**

Οι **MFCCs** είναι αριθμητικές τιμές που αναπαριστούν **χαρακτηριστικά του φάσματος της φωνής** με τρόπο που **προσομοιώνει την ανθρώπινη ακοή**. Είναι ιδιαίτερα χρήσιμοι στην αναγνώριση φωνής και συναισθημάτων.

**🧩 Βήματα Παραγωγής MFCCs**

Ακολουθεί η πλήρης αλυσίδα επεξεργασίας ενός ηχητικού σήματος για την εξαγωγή των MFCCs:

**✅ Βήμα 1: Pre-Emphasis**

Εφαρμόζεται ένα φίλτρο για την ενίσχυση των υψηλών συχνοτήτων.

**Γιατί;** Επειδή η ανθρώπινη φωνή έχει λιγότερη ενέργεια στις υψηλές συχνότητες, και θέλουμε να τις αναδείξουμε για καλύτερη ανάλυση.

🔹 Συνήθης εξίσωση:  
y[n]=x[n]−α⋅x[n−1]y[n] = x[n] - \alpha \cdot x[n-1] με α≈0.95\alpha ≈ 0.95

**✅ Βήμα 2: Framing (Κατακερματισμός σε πλαίσια)**

Το σήμα χωρίζεται σε **μικρά κομμάτια (frames)** των 20–40 ms, π.χ. 25ms με επικάλυψη 10ms.

🔹 Γιατί; Το σήμα θεωρείται **σταθερό (stationary)** μόνο για μικρό χρονικό διάστημα.

**✅ Βήμα 3: Windowing**

Εφαρμόζεται **παράθυρο (συνήθως Hamming)** σε κάθε frame για να αποφευχθούν "κοψίματα" στα άκρα.

🔹 Συνάρτηση Hamming:  
w[n]=0.54−0.46⋅cos⁡(2πnN−1)w[n] = 0.54 - 0.46 \cdot \cos\left( \frac{2\pi n}{N-1} \right)

**✅ Βήμα 4: Fast Fourier Transform (FFT)**

Μετατρέπει το σήμα από χρονικό σε **συχνοτικό πεδίο**.

🔹 Παίρνουμε το **φάσμα συχνοτήτων** κάθε frame.

**✅ Βήμα 5: Mel Filter Bank**

Πολλαπλασιάζουμε το φάσμα με ένα σύνολο από **τριγωνικά φίλτρα** κατανεμημένα στη **Mel κλίμακα**.

🔹 Η **Mel κλίμακα** μοντελοποιεί το πώς **αντιλαμβανόμαστε** τις συχνότητες:

* Κάτω από τα ~1000 Hz: γραμμική
* Πάνω από τα ~1000 Hz: λογαριθμική

🔸 Μετατροπή σε Mel:

m=2595⋅log⁡10(1+f700)m = 2595 \cdot \log\_{10}\left(1 + \frac{f}{700}\right)

Παράδειγμα: αν έχεις 40 φίλτρα, καταλήγεις σε 40 ενεργειακές τιμές (μία για κάθε φίλτρο).

**✅ Βήμα 6: Logarithm**

Εφαρμόζουμε **λογάριθμο** στις εντάσεις των φίλτρων για να προσομοιώσουμε την **αντίληψη έντασης** από το ανθρώπινο αυτί (logarithmic loudness perception).

**✅ Βήμα 7: Discrete Cosine Transform (DCT)**

Αντί να δουλέψουμε με τις ενεργειακές τιμές φίλτρων, εφαρμόζουμε DCT και κρατάμε τους **πρώτους 12–13 συντελεστές**.

🔹 Γιατί DCT;  
Μετατρέπει τις τιμές σε **ασυσχέτιστους συντελεστές**, βοηθώντας στη **συμπίεση πληροφορίας** και την **ταξινόμηση**.

**✅ (Προαιρετικά) Βήμα 8: Διαφορικές συναρτήσεις (Δέλτα, Δέλτα-Δέλτα)**

* **Δέλτα (Δ')**: Ταχύτητα αλλαγής MFCC.
* **Δέλτα-Δέλτα (Δ'')**: Επιτάχυνση της αλλαγής.

Προσθέτουν **χρονικό δυναμισμό** στο χαρακτηριστικό.

**🧪 Τελικό αποτέλεσμα:**

Για κάθε frame (π.χ. κάθε 25ms), παίρνουμε π.χ. 13 MFCCs → έτσι προκύπτει ένα **μήτρα χαρακτηριστικών MFCCs** με μορφή:

[ [MFCC1\_frame1, ..., MFCC13\_frame1],

[MFCC1\_frame2, ..., MFCC13\_frame2],

...

]

**📌 Παράδειγμα με τη Librosa (Python)**

import librosa

y, sr = librosa.load("audio.wav")

mfccs = librosa.feature.mfcc(y=y, sr=sr, n\_mfcc=13)

**📊 Οπτικοποίηση:**

Αν απεικονίσεις τους MFCCs σαν εικόνα (με χρόνο στον άξονα x και MFCC δείκτη στον άξονα y), έχεις ένα "θερμικό αποτύπωμα" του ήχου, όπως στην **Figure 2** του άρθρου που ανέλυσες.

Θες να σου δείξω ένα **διάγραμμα** της διαδικασίας παραγωγής MFCCs ή να σε βοηθήσω να το γράψεις ως ροή εργασίας για αναφορά/παρουσίαση;

Στο άρθρο όταν αναφέρεται στο ότι κρατάει τα πρώτα 40 MFCCs εννοεί ότι χωρίζει το σήμα ήχου σε 40 πλαίσια στο βήμα 2 ή πολλαπλασιάζει το σήμα με 40 φίλτρα στο βήμα 5 και άρα έχει 40 ζώνες συχνότητας ή και τα δύο. Και αν κρατάμε 40 MFCCs στο βήμα 4 και 7 που κάνουμε τους μετασχηματισμούς απο αυτούς πόσους συντελεστές κρατάμε, 40 όσα και τα MFCCs;

Επίσης όταν λες ότι λογαριθμούμε στο βήμα 6 εννοείς ότι εφαρμόζουμε A-weighting στο σήμα για να είναι σε dB(A) ή ότι βάζουμε το φίλτρο Equal-loudness contour για να προσομοιώσουμε την ευαισθησία του αυτιού του ανθρώπου

Πολύ καλές και στοχευμένες ερωτήσεις — πάμε να τις απαντήσουμε προσεκτικά:

**🔹 Όταν το άρθρο λέει "κρατάμε τους πρώτους 40 MFCCs", τι εννοεί;**

**Δεν εννοεί 40 πλαίσια** (frames) του σήματος.  
Αντίθετα, εννοεί ότι **κρατάει 40 φίλτρα στη Mel κλίμακα** και εξάγει **40 MFCCs ανά frame**.

**🔸 Αναλυτικά:**

* **Βήμα 2**: Το σήμα χωρίζεται σε πολλά **frames** (π.χ. κάθε 25ms με επικάλυψη).
* **Βήμα 5**: Το φάσμα κάθε frame φιλτράρεται με **40 τριγωνικά φίλτρα στη Mel κλίμακα** → δίνει **40 τιμές ισχύος**.
* **Βήμα 6**: Λογάριθμος αυτών των 40 τιμών (log-energy per band).
* **Βήμα 7**: Εφαρμόζεται **Discrete Cosine Transform (DCT)** στις 40 log-ενέργειες.
* Συνήθως **κρατάμε τους πρώτους 12–13 DCT coefficients** ως **τελικούς MFCCs**, αλλά στο άρθρο κρατούν **και τους 40** (δηλ. όλη τη σειρά των DCT coefficients).

🔸 **Άρα:**

* 40 φίλτρα ➜ 40 log-ενεργειακές τιμές ➜ DCT ➜ **40 MFCCs**
* Και το άρθρο **χρησιμοποιεί όλους τους 40 MFCCs ανά frame**, όχι μόνο τους πρώτους 12–13.

**🔹 Σχετικά με το λογάριθμο στο Βήμα 6 — Είναι A-weighting ή Equal-loudness contour;**

Όχι, **ούτε A-weighting**, ούτε equal-loudness filters εφαρμόζονται εδώ.

**🔸 Τι εννοούμε με "λογάριθμο" στο Βήμα 6:**

Εφαρμόζεται **απλός φυσικός ή δεκαδικός λογάριθμος**:

log-energy=log⁡(E)\text{log-energy} = \log(E)

όπου EE είναι η ενέργεια κάθε φίλτρου στη Mel κλίμακα.

**🔸 Γιατί εφαρμόζουμε log;**

* Ο **λογαριθμικός μετασχηματισμός**:
  + Μιμείται την **αντίληψη έντασης** του ανθρώπινου αυτιού (π.χ. ο διπλασιασμός έντασης δεν ακούγεται ως "διπλάσιος").
  + Βοηθάει στον **εξισορροπημένο χειρισμό πολύ υψηλών και πολύ χαμηλών τιμών**.

**🔸 A-weighting και Equal-loudness filters:**

Αυτά είναι **άλλα φίλτρα** που εφαρμόζονται για να μοντελοποιήσουν **την ευαισθησία της ακοής ανά συχνότητα**.

* **A-weighting**: πιο χρήσιμο για **ηχητική μέτρηση περιβάλλοντος** (π.χ. dB(A)).
* **Equal-loudness contours** (ISO 226): μοντελοποιούν **πώς αντιλαμβανόμαστε την ένταση στις διάφορες συχνότητες**, **ανάλογα με τη στάθμη**.

👉 **Κανένα από τα δύο δεν χρησιμοποιείται τυπικά στη διαδικασία εξαγωγής MFCCs**. Η **προσομοίωση της ακουστικής αντίληψης** εδώ γίνεται κυρίως:

* μέσω της **Mel κλίμακας** (βήμα 5),
* και της **λογάριθμης μετατροπής** (βήμα 6).

**✅ Συνοψίζοντας:**

| **Βήμα** | **Τι σημαίνει "40 MFCCs" στο άρθρο** |
| --- | --- |
| **Βήμα 2** | Όχι 40 πλαίσια — το σήμα χωρίζεται σε πολλά frames (π.χ. 100+) |
| **Βήμα 5** | Χρησιμοποιούνται 40 φίλτρα στη Mel κλίμακα |
| **Βήμα 7** | Εφαρμόζεται DCT σε 40 log-ενέργειες ➜ προκύπτουν 40 MFCCs |
| **Τελικό** | Κρατούν **και τους 40 MFCCs** (όχι μόνο τους 12 πρώτους) |

| **Βήμα 6** | **Εφαρμογή log** |
| --- | --- |
| Τύπος | log⁡(energy)\log(\text{energy}), όχι A-weighting ή equal-loudness |
| Σκοπός | Προσομοίωση ανθρώπινης αντίληψης έντασης |

Αν θέλεις, μπορώ να σου φτιάξω **σχηματικό διάγραμμα** (flowchart ή μπλοκ διάγραμμα) με όλα τα βήματα της εξαγωγής MFCCs. Θες να το ετοιμάσω;