



ΠΑΝΕΠΙΣΤΗΜΙΟ  
ΘΕΣΣΑΛΙΑΣ

ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ

ΜΕΛΕΤΗ ΚΑΙ ΕΦΑΡΜΟΓΗ ΤΟΥ ΦΙΛΤΡΟΥ  
SOBEL ΣΤΟ ΝΕΟ ΠΡΟΤΥΠΟ ΣΥΜΠΙΕΣΗΣ  
VIDEO VVC

ΤΣΕΠΕΛΙΔΗΣ ΓΕΩΡΓΙΟΣ

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

ΥΠΕΥΘΥΝΟΣ

Κοζύρη Μαρία  
Επικουρη Καθηγήτρια

Λαμία ..... έτος ...2021...





ΠΑΝΕΠΙΣΤΗΜΙΟ  
ΘΕΣΣΑΛΙΑΣ

ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ

ΜΕΛΕΤΗ ΚΑΙ ΕΦΑΡΜΟΓΗ ΤΟΥ ΦΙΛΤΡΟΥ  
SOBEL ΣΤΟ ΝΕΟ ΠΡΟΤΥΠΟ ΣΥΜΠΙΕΣΗΣ  
VIDEO VVC

ΤΣΕΠΕΛΙΔΗΣ ΓΕΩΡΓΙΟΣ

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

ΥΠΕΥΘΥΝΟΣ

Κοζύρη Μαρία  
Επικουρη Καθηγήτρια

Λαμία ..... έτος ...2021...





UNIVERSITY OF  
THESSALY

SCHOOL OF SCIENCE

DEPARTMENT OF COMPUTER SCIENCE & TELECOMMUNICATIONS

RESEARCH AND IMPLEMENTATION OF  
SOBEL FILTER IN THE NOVEL VIDEO  
COMPRESSION STANDARD VVC

TSEPELIDIS GEORGIOS

FINAL THESIS

ADVISOR

Koziri Maria  
Assistant Professor

Lamia ..... year ...2021...



«Με ατομική μου ευθύνη και γνωρίζοντας τις κυρώσεις <sup>(1)</sup>, που προβλέπονται από της διατάξεις της παρ. 6 του άρθρου 22 του Ν. 1599/1986, δηλώνω ότι:

1. Δεν παραθέτω κομμάτια βιβλίων ή άρθρων ή εργασιών άλλων αυτολεξεί **χωρίς να τα περικλείω σε εισαγωγικά** και χωρίς να αναφέρω το συγγραφέα, τη χρονολογία, τη σελίδα. Η αυτολεξεί παράθεση χωρίς εισαγωγικά χωρίς αναφορά στην πηγή, είναι λογοκλοπή. Πέραν της αυτολεξεί παράθεσης, λογοκλοπή θεωρείται και η παράφραση εδαφίων από έργα άλλων, συμπεριλαμβανομένων και έργων συμφοιτητών μου, καθώς και η παράθεση στοιχείων που άλλοι συνέλεξαν ή επεξεργάστηκαν, χωρίς αναφορά στην πηγή. Αναφέρω πάντοτε με πληρότητα την πηγή κάτω από τον πίνακα ή σχέδιο, όπως στα παραθέματα.

2. Δέχομαι ότι η αυτολεξεί **παράθεση χωρίς εισαγωγικά**, ακόμα κι αν συνοδεύεται από αναφορά στην πηγή σε κάποιο άλλο σημείο του κειμένου ή στο τέλος του, είναι αντιγραφή. Η αναφορά στην πηγή στο τέλος π.χ. μιας παραγράφου ή μιας σελίδας, δεν δικαιολογεί συρραφή εδαφίων έργου άλλου συγγραφέα, έστω και παραφρασμένων, και παρουσίασή τους ως δική μου εργασία.

3. Δέχομαι ότι υπάρχει επίσης περιορισμός στο μέγεθος και στη συχνότητα των παραθεμάτων που μπορώ να εντάξω στην εργασία μου εντός εισαγωγικών. Κάθε μεγάλο παράθεμα (π.χ. σε πίνακα ή πλαίσιο, κλπ), προϋποθέτει ειδικές ρυθμίσεις, και όταν δημοσιεύεται προϋποθέτει την άδεια του συγγραφέα ή του εκδότη. Το ίδιο και οι πίνακες και τα σχέδια

4. Δέχομαι όλες τις συνέπειες σε περίπτωση λογοκλοπής ή αντιγραφής.

Ημερομηνία: ...../...../2021...

Ο – Η Δηλ.

(1) «Όποιος εν γνώσει του δηλώνει ψευδή γεγονότα ή αρνείται ή αποκρύπτει τα αληθινά με έγγραφη υπεύθυνη δήλωση του άρθρου 8 παρ. 4 Ν. 1599/1986 τιμωρείται με φυλάκιση τουλάχιστον τριών μηνών. Εάν ο υπαίτιος αυτών των πράξεων σκόπευε να προσπορίσει στον εαυτόν του ή σε άλλον περιουσιακό όφελος βλάπτοντας τρίτον ή σκόπευε να βλάψει άλλον, τιμωρείται με κάθειρξη μέχρι 10 ετών.»







## ΠΕΡΙΛΗΨΗ

---

Σε αυτή τη πτυχιακή εργασία γίνεται μία αναφορά στις τεχνικές συμπίεσης βίντεο και στα πρότυπα τους καθώς και μία πρόταση ενός νέου τρόπου μείωσης της Intra-frame πολυπλοκότητας στο νέο πρότυπο συμπίεσης βίντεο VVC χρησιμοποιώντας το κριτήριο διακύμανσης. Στο δεύτερο κεφάλαιο γίνεται μία επισκόπηση της βιβλιογραφίας. Το τρίτο κεφάλαιο αναλύει τις τεχνικές συμπίεσης βίντεο και τα πρότυπά τους ενώ στο τέταρτο κεφάλαιο περιγράφεται πλήρως ο νέος τρόπος συμπίεσης Intra-frame. Τέλος, στο πέμπτο κεφάλαιο παρουσιάζονται τα αποτελέσματα από την υλοποίηση και εφαρμογή του νέου τρόπου συμπίεσης βίντεο στο VVC Test Model (VTM) της ομάδας Joint Video Experts Team (JVET).



## ABSTRACT

---

In this thesis, video compression techniques and their standards are referenced as well as a new manner of Intra-frame complexity reduction in the novel video compression standard VVC is proposed utilizing the variance criterion. In the second chapter there is an overview of the bibliography. The third chapter analyses video compression techniques and their standards while in the fourth chapter the new manner of Intra-frame compression is described thoroughly. Finally, in the fifth chapter the results from the implementation and application of the novel way of video compression in the VVC Test Model (VTM) from the Joint Video Experts Team (JVET) are presented.





## Table of Contents

---

ΠΕΡΙΛΗΨΗ .....	I
ABSTRACT .....	III
<b><u>ΚΕΦΑΛΑΙΟ 1 ΕΙΣΑΓΩΓΗ .....</u></b>	<b><u>1</u></b>
ΓΕΝΙΚΕΣ ΠΛΗΡΟΦΟΡΙΕΣ ΓΙΑ ΤΗ ΣΥΜΠΙΕΣΗ ΚΑΙ ΤΑ ΠΡΟΤΥΠΑ .....	1
ΤΥΠΙΚΗ ΔΟΜΗ ΕΝΟΣ ΠΡΟΤΥΠΟΥ ΣΥΜΠΙΕΣΗΣ ΒΙΝΤΕΟ .....	3
ΣΥΜΠΙΕΣΗ INTRA ΚΑΙ INTER-FRAME .....	6
<b><u>ΚΕΦΑΛΑΙΟ 2 ΒΙΒΛΙΟΓΡΑΦΙΚΗ ΕΠΙΣΚΟΠΗΣΗ .....</u></b>	<b><u>8</u></b>
<b><u>ΚΕΦΑΛΑΙΟ 3 ΠΡΟΤΥΠΟ ΣΥΜΠΙΕΣΗΣ VIDEO VVC .....</u></b>	<b><u>9</u></b>
ΠΡΟΤΥΠΟ ΚΩΔΙΚΟΠΟΙΗΣΗΣ ΒΙΝΤΕΟ VVC/H.266.....	9
ΚΑΙΝΟΥΡΓΙΑ ΧΑΡΑΚΤΗΡΙΣΤΙΚΑ ΣΤΟ ΠΡΟΤΥΠΟ VVC/H.266.....	9
ΕΙΣΑΓΩΓΗ ΔΕΝΤΡΟΥ ΠΟΛΛΑΠΛΩΝ ΤΥΠΩΝ ΚΑΙ ΝΕΑ ΜΕΓΕΘΗ CTU.....	9
INTRA PREDICTION.....	10
INTER PREDICTION.....	13
<b><u>ΚΕΦΑΛΑΙΟ 4 ΑΝΙΧΝΕΥΣΗ ΑΚΜΩΝ .....</u></b>	<b><u>15</u></b>
ΠΛΗΡΟΦΟΡΙΕΣ ΓΙΑ ΤΗΝ ΑΝΙΧΝΕΥΣΗ ΑΚΜΩΝ.....	15
ΤΕΛΕΣΤΗΣ SOBEL.....	16
<b><u>ΚΕΦΑΛΑΙΟ 5 VVC TEST MODEL .....</u></b>	<b><u>18</u></b>
ΕΙΣΑΓΩΓΙΚΟ ΣΗΜΕΙΩΜΑ .....	18
VTM, ΕΚΤΕΛΕΣΗ ΚΑΙ ΡΟΗ ΤΟΥ ΚΩΔΙΚΟΠΟΙΗΤΗ .....	18
ΡΟΗ ΚΑΙ ΤΡΟΠΟΣ ΛΕΙΤΟΥΡΓΙΑΣ ΣΥΜΠΙΕΣΗΣ INTRA-FRAME ΣΤΟ VTM.....	19
ΔΟΜΗ ΤΟΥ ΑΛΓΟΡΙΘΜΟΥ.....	20
Ο ΑΛΓΟΡΙΘΜΟΣ.....	20
<b><u>ΚΕΦΑΛΑΙΟ 6 ΠΕΙΡΑΜΑΤΙΚΑ ΑΠΟΤΕΛΕΣΜΑΤΑ .....</u></b>	<b><u>28</u></b>
<b><u>ΚΕΦΑΛΑΙΟ 7 ΣΥΜΠΕΡΑΣΜΑΤΑ.....</u></b>	<b><u>36</u></b>
<b><u>ΞΕΝΗ ΒΙΒΛΙΟΓΡΑΦΙΑ.....</u></b>	<b><u>37</u></b>

# ΚΕΦΑΛΑΙΟ 1 Εισαγωγή

---

## Γενικές πληροφορίες για τη συμπίεση και τα πρότυπα

---

Με τον όρο συμπίεση εννοούμε την αφαίρεση πλεονάζουσας πληροφορίας από οποιοδήποτε ψηφιακό αρχείο έτσι ώστε να μειωθεί το αρχικό μέγεθός του. Όσον αφορά τη συμπίεση ψηφιακής εικόνας και βίντεο, αυτή χωρίζεται σε δύο διαφορετικά είδη ανάλογα με την επίδραση που έχει η συμπίεση στη ποιότητα της εικόνας και του βίντεο. Χωρίζεται στην απωλεστική (lossy) και στη μη απωλεστική (lossless) συμπίεση. Αξίζει να σημειωθεί ότι οι δύο προηγούμενες μορφές συμπίεσης ισχύουν και για τον ήχο.

Στην απωλεστική συμπίεση ή συμπίεση με απώλειες ένα ποσοστό δεδομένων από το αρχικό αρχείο χάνεται κατά τη διάρκεια της συμπίεσης. Επομένως, ένα βίντεο ή ένας ήχος που έχει συμπιεστεί με μία απωλεστική μέθοδο συμπίεσης κατά την αναπαραγωγή του θα διαφέρει από το αρχικό αρχείο στη ποιότητα εικόνας και ήχου αντίστοιχα.

Στη μη απωλεστική συμπίεση ή συμπίεση χωρίς απώλειες δεν χάνεται πληροφορία ενός αρχείου κατά τη διάρκεια της συμπίεσης. Έτσι, στην αποκωδικοποίηση ενός ήχου ή μίας εικόνας, η μέθοδος μη απωλεστικής συμπίεσης θα επαναφέρει το αρχικό αρχείο, χωρίς να υπάρχει κάποια διαφορά στη ποιότητα, είναι, δηλαδή, μία πλήρως αντιστρέψιμη διαδικασία.

Ανέκαθεν ο όγκος της πληροφορίας που παράγει ένα βίντεο ήταν αρκετά μεγάλος για να μπορέσει να μεταδοθεί μέσω ενός οποιοδήποτε ψηφιακού σήματος πόσο μάλλον να καταφέρει να επεξεργαστεί στον δέκτη στον οποίο καταλήγει. Επιπλέον, υπάρχει η ανάγκη για μετάδοση αυτής της πληροφορίας μέσα από μία μέση σύνδεση στο διαδίκτυο όπως για τη χρήση βιντεοτηλεφωνίας (videotelephony) ή βιντεοδιασκέψεων (videoconferencing). Επομένως, εμφανίζεται η ανάγκη για τη κωδικοποίηση της πληροφορίας που παράγει ένα βίντεο έτσι ώστε να μπορεί να μεταδοθεί όχι μόνο από ένα ψηφιακό σήμα αλλά και μέσω διαδικτύου καθώς και για τη σωστή επεξεργασία της ροής πληροφορίας που καταλήγει σε κάποιο αποδέκτη μέσω της αποκωδικοποίησης.

Το καθήκον ανάπτυξης προτύπων συμπίεσης βίντεο έχουν αναλάβει συγκεκριμένες ομάδες ειδικών από διάφορες διεθνείς ενώσεις και οργανισμούς οι οποίοι συνεργάζονται μεταξύ τους για τη τυποποίηση προτύπων που αφορούν τη κωδικοποίηση βίντεο. Παρακάτω ακολουθούν σύντομες περιγραφές των οργανισμών ISO, IEC, ITU καθώς και των ομάδων ειδικών που έχουν δημιουργηθεί από τη συνεργασία τους για την ανάπτυξη των προτύπων.

Ο Διεθνής Οργανισμός Τυποποίησης ή ISO (International Organization for Standardization) είναι ένας διεθνής οργανισμός στον οποίο αναπτύσσονται και δημοσιεύονται διεθνή πρότυπα [4] που αφορούν τη διαχείριση της ποιότητας προϊόντων, πρότυπα για υγεία και ασφάλεια στο χώρο εργασίας, πρότυπα για τη σωστή διαχείριση των τροφίμων, πρότυπα για τη συνεισφορά στη προστασία του



περιβάλλοντος μεταξύ άλλων [5]. Ο οργανισμός ιδρύθηκε το 1947 και έχει ως μέλη 165 εθνικούς οργανισμούς τυποποιήσεων με τους οποίους συνεργάζεται [4][6]. Η Διεθνής Ηλεκτροτεχνική Επιτροπή ή IEC (International Electrotechnical Commission) είναι μία διεθνής επιτροπή στην οποία αναπτύσσονται και δημοσιεύονται διεθνή πρότυπα για τη τυποποίηση ηλεκτρικών και ηλεκτρονικών τεχνολογιών [11]. Το JTC 1 πρόκειται για μία τεχνική επιτροπή που δημιουργήθηκε από τον οργανισμό ISO και την επιτροπή IEC για τις τυποποιήσεις προτύπων που αφορούν το τομέα της Πληροφορικής [7]. Το Moving Picture Experts Group (MPEG) ανήκει σε αυτή τη τεχνική επιτροπή και είναι υπεύθυνο για τη τυποποίηση προτύπων σε ό,τι αφορά την κωδικοποίηση της πληροφορίας από τον ήχο, την εικόνα, τα πολυμέσα και τα υπερμέσα [8]. Η Διεθνής Ένωση Τηλεπικοινωνιών ή ITU (International Telecommunications Unit) είναι μία ειδική ένωση των Ηνωμένων Εθνών που ιδρύθηκε το 1865 με σκοπό να ενώσει το κόσμο δημιουργώντας πρότυπα σε τεχνολογίες πληροφορικής και επικοινωνιών [10]. Η ομάδα Video Coding Experts Group (VCEG) είναι μία ομάδα ειδικών στη κωδικοποίηση εικόνας και βίντεο από την οποία γίνονται συστάσεις για πρότυπα συμπίεσης βίντεο και εικόνας [11]. Η JVET είναι μία ομάδα ειδικών στη κωδικοποίηση βίντεο από τις ομάδες VCEG του ITU και MPEG της τεχνικής επιτροπής JTC 1 του ISO/IEC και δημιουργήθηκε τον Οκτώβριο του 2017 με σκοπό τη δημιουργία ενός νέου προτύπου συμπίεσης βίντεο με όνομα Versatile Video Coding (VVC) [9]. Το 2020 τυποποιήθηκε η πρώτη εκδοχή του προτύπου με αριθμούς H.266 και ISO/IEC 23090-3 από ένωση ITU και την ένωση των ISO και IEC αντίστοιχα [2][3].

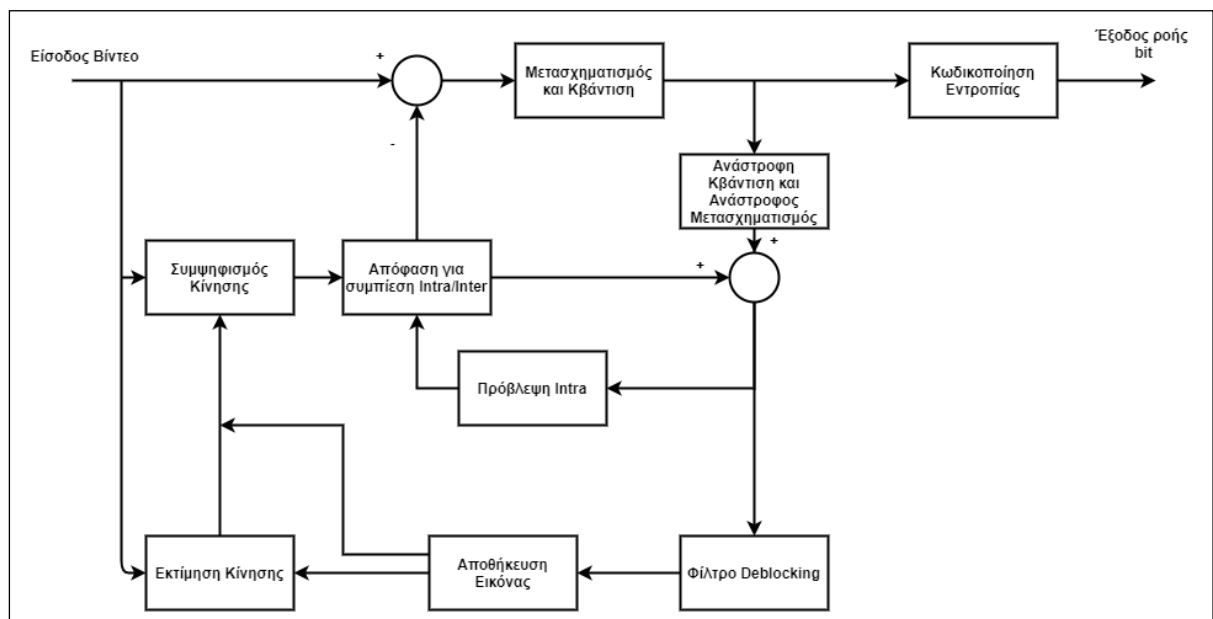
Το λογισμικό ή οι συσκευές οι οποίες κωδικοποιούν ένα ψηφιακό βίντεο και έπειτα το αποκωδικοποιούν ονομάζονται κωδικοποιητές ή κωδικοποιητές βίντεο και χρησιμοποιούνται στις ψηφιακές τηλεοράσεις, για τη προβολή βίντεο μέσω διαδικτύου, στη ζωντανή μετάδοση, στις βιντεοδιασκέψεις και άλλα. Οι κωδικοποιητές, είτε είναι σε μορφή λογισμικού είτε υλικού, αναπτύσσονται από τις διάφορες εταιρίες βάσει τυποποιημένων προτύπων, δημιουργώντας κατ' αυτόν το τρόπο κωδικοποιητές συμβατούς με ένα μεγάλο πλήθος συσκευών και έτοιμους να δεχθούν ένα τηλεπικοινωνιακό σήμα με σκοπό τη προβολή σε οποιαδήποτε συσκευή.

Κάποια γνωστά πρότυπα κωδικοποιητών που χρησιμοποιούνται ευρέως σήμερα για τη συμπίεση βίντεο είναι τα Advanced Video Codec (AVC) ή H.264 και High Efficiency Video Codec (HEVC) ή H.265 και είναι το αποτέλεσμα συνεργασίας μεταξύ των προαναφερθέντων οργανισμών, δηλαδή των ISO, IEC και ITU-T. Επιπλέον, σύμφωνα με την Ετήσια Αναφορά Προγραμματιστών Βίντεο της Bitmovin του 2019 [14], το πρότυπο AVC έχει το μεγαλύτερο ποσοστό υιοθέτησης στην αγορά από για καταγραφή, συμπίεση και διανομή βίντεο, φτάνοντας το 91% σε μερίδιο υιοθέτησης. Ενώ το αμέσως επόμενο πρότυπο, το HEVC/H.265, χρησιμοποιείται από το 43% των προγραμματιστών βίντεο. Το νεότερο πρότυπο είναι το Versatile Video Coding (VVC) ή H.266 το οποίο βρισκόταν υπό ανάπτυξη από το 2017 από τους παραπάνω οργανισμούς και τυποποιήθηκε τον Ιούλιο του 2020. Σε αυτή τη πτυχιακή θα ασχοληθούμε με το νεότερο πρότυπο κάνοντας μία αναφορά στα νέα χαρακτηριστικά που προσφέρει σε σχέση με τα προηγούμενα πρότυπα καθώς και η παρουσίαση ενός αλγορίθμου ο οποίος έχει ως στόχο τη μείωση της πολυπλοκότητας χρησιμοποιώντας το κριτήριο της διακύμανσης στη συμπίεση

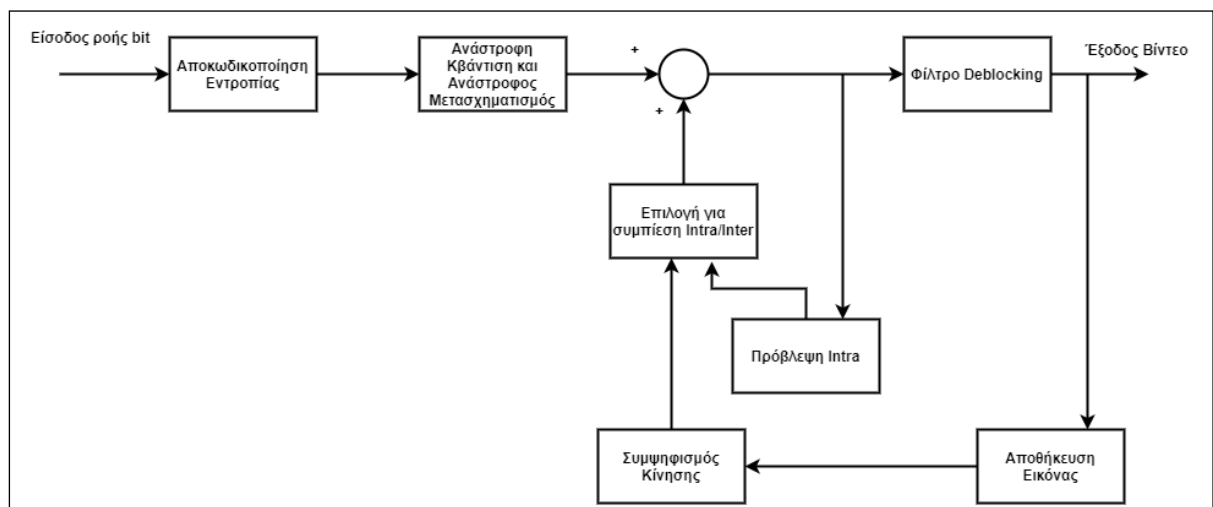
Intra-frame του προτύπου. Στα επόμενα κεφάλαια αναλυθεί πλήρως ο αλγόριθμος καθώς και οι αλλαγές στο κώδικα.

## Τυπική δομή ενός προτύπου συμπίεσης βίντεο

Τα πρότυπα συμπίεσης βίντεο όσο και να διαφέρουν από άποψη απόδοσης, υπολογιστικής πολυπλοκότητας, χρόνου εκτέλεσης και άλλα, ακολουθούν ένα παρόμοιο τρόπο στη διαδικασία της κωδικοποίησης και της αποκωδικοποίησης. Αυτή η διαδικασία πραγματοποιείται μέσω ενός υβριδικού μοντέλου κωδικοποίησης και αποκωδικοποίησης. Στις παρακάτω εικόνες φαίνονται τα διαγράμματα ροής ενός υβριδικού μοντέλου κωδικοποιητή και αποκωδικοποιητή που έχει υιοθετηθεί από τα περισσότερα πρότυπα.



Εικόνα 1. Διάγραμμα ροής Υβριδικού Μοντέλου κωδικοποιητή.



Εικόνα 2. Διάγραμμα ροής Υβριδικού Μοντέλου αποκωδικοποιητή.

Ένα καρτέ ενός βίντεο, ή καλύτερα μία εικόνα, και κατά συνέπεια ένα ολόκληρο βίντεο πρέπει να περάσει από κάποιες συγκεκριμένες διαδικασίες με μερικές από αυτές να είναι γνωστές και από τη ψηφιακή επεξεργασία εικόνας.

Αρχικά, μία εικόνα πρέπει να χωριστεί στα μπλοκ κωδικοποίησης σε μεγέθη που επιτρέπει το πρότυπο. Τα μπλοκ αυτά ονομάζονται Coding Tree Units στα καινούργια πρότυπα ενώ σε προηγούμενα πρότυπα ονομαζόντουσαν Macroblocks και αποτελούν τη βασική μονάδα κωδικοποίησης. Τα μπλοκ αυτά χωρίζονται περαιτέρω αναλόγως, και πάλι, με το πρότυπο συμπίεσης, καθώς τα νεότερα πρότυπα αυξάνουν το μέγιστο μέγεθος των μπλοκ και επιτρέπουν τη εφαρμογή νέων τρόπων διαχωρισμού με χρήση μπλοκ διαφορετικών σχημάτων.

Έπειτα, τα μπλοκ αυτά και κατ' επέκταση ολόκληρη η εικόνα πρέπει να περάσει από τη διαδικασία του μετασχηματισμού. Στο μετασχηματισμό, μία εικόνα, με τη χρήση κάποιας συνάρτησης μετασχηματισμού, περνάει από το χωρικό πεδίο ορισμού στο πεδίο ορισμού της συχνότητας όπου η πλεονάζουσα πληροφορία μπορεί να αναγνωρισθεί και με τη βοήθεια των επόμενων διαδικασιών να αφαιρεθεί. Ο μετασχηματισμός από μόνος του προσφέρει ένα βαθμό συμπίεσης αλλά συνήθως συνδυάζεται με άλλες διαδικασίες για να επιτευχθεί ένας μεγαλύτερος βαθμός συμπίεσης. Στη συμπίεση βίντεο, όπως και στη συμπίεση ψηφιακών εικόνων, συμπίεση ήχου, επεξεργασία σημάτων και άλλα, ο μετασχηματισμός που έχει επιλεγεί είναι ο διακριτός συνημίτονος μετασχηματισμός ή Discrete Cosine Transform (DCT) για τη μεταφορά στο πεδίο ορισμού της συχνότητας καθώς και για τη συμπίεση που μπορεί να προσφέρει.

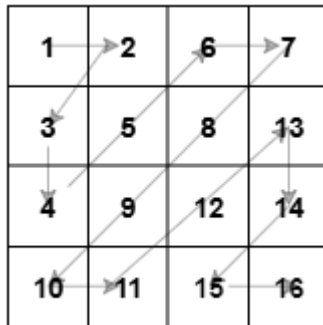
Στη συνέχεια, η εικόνα που αποτελεί το αποτέλεσμα του μετασχηματισμού θα πρέπει να περάσει από τη διαδικασία του κβαντισμού, μία τεχνική που ανήκει στην απωλεστική συμπίεση και έχει εφαρμογές στη ψηφιακή επεξεργασία εικόνας. Στο κβαντισμό, το εύρος τιμών που έχουν τα εικονοστοιχεία από το μετασχηματισμό θα μετατραπεί σε ένα πλήθος βημάτων (steps) μικρότερο από το εύρος τιμών χρησιμοποιώντας ένα πίνακα κβαντισμού (quantization matrix) με σκοπό να μειώσει τις τιμές που θα πρέπει να περάσουν από κάποιο μέσο συνεχούς ροής (stream). Στη παρακάτω εικόνα φαίνεται ένας συνηθισμένος πίνακας κβαντισμού που χρησιμοποιείται στο μετασχηματισμένο στοιχείο της φωτεινότητας.

16	11	10	16	24	40	51	61
12	12	14	19	26	58	60	55
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

Εικόνα 3. Ένας 8x8 πίνακας κβαντισμού.

Μετά την εφαρμογή ενός πίνακα κβαντισμού στο μετασχηματισμένο μπλοκ τα μπλοκ καταλήγουν να έχουν ένα μεγάλο πλήθος μηδενικών στο οποίο μπορεί να

εφαρμοστεί μία μορφή συμπίεσης. Ο κωδικοποιητής διασχίζει το μπλοκ σε ένα μοτίβο zig-zag (zig-zag traversal) και συμπιέζει τις σειρές μηδενικών που συναντάει, αντικαθιστώντας τα μηδενικά με ένα μηδενικό και τον αριθμό συνολικών μηδενικών τα οποία ήταν σε συστοιχία. Η συμπίεση αυτή ονομάζεται Run-length Encoding (RLE) και είναι μία μορφή μη απωλεστικής συμπίεσης που χρησιμοποιείται σε διάφορες απλές εικόνες όπως και στη συμπίεση βίντεο.



Εικόνα 4. Η διάσχιση zig-zag, με τη συστοιχία να είναι 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16.

Επιπλέον υπάρχουν δύο διαδικασίες που χρησιμοποιούνται στο υβριδικό μοντέλο κωδικοποίησης και αποτελούν σημαντικό μέρος της Inter-frame πρόβλεψης είναι η Εκτίμηση Κίνησης (Motion Estimation) και ο Συμψηφισμός Κίνησης (Motion Compensation). Στη διαδικασία της Εκτίμησης Κίνησης ο κωδικοποιητής ελέγχοντας τη τωρινή εικόνα καθώς και τη προηγούμενη και υπολογίζει τα διανύσματα κίνησης (motion vectors). Τα διανύσματα κίνησης χρησιμοποιούνται για να δείξουν τη κίνηση των μπλοκ από τη μία εικόνα στην επόμενη και αποτελούν μέρος στο Συμψηφισμό Κίνησης. Στη διαδικασία του Συμψηφισμού Κίνησης ο κωδικοποιητής αποθηκεύει της αλλαγές της τωρινής εικόνας με μία εικόνα που έχει επιλέξει σαν αναφορά για αυτή η οποία ονομάζεται εικόνα αναφοράς (reference picture). Η εικόνα αναφοράς μπορεί να προέλθει είτε από προηγούμενη εικόνα είτε από επόμενη και επιτυγχάνεται με τη δομή GOP (Group of Pictures). Αναφορά για τη δομή GOP όπως και για τη πρόβλεψη Inter-frame θα γίνει σε παρακάτω παραγράφους.

Αυτές οι διαδικασίες αποτελούν μέρος της κωδικοποίησης και στο τέλος της η εικόνα έχει συμπεστεί, χρειάζεται όμως να αποκωδικοποιηθεί. Για την αποκωδικοποίηση η εικόνα πρέπει να περάσει από τις προηγούμενες διαδικασίες με αντίθετη φορά. Όλες οι διαδικασίες της κωδικοποίησης είναι αντιστρέψιμες όπως η κβάντιση με την αντιστροφή κβάντιση (inverse quantization) και το διακριτό συνημίτονο μετασχηματισμό με την Inverse DCT με κάποιες διαδικασίες να είναι πλήρως αντιστρέψιμες όπως τη κωδικοποίηση εντροπίας που ανήκει στη μη απωλεστική μορφή συμπίεσης.

Πέρα από το μετασχηματισμό, τη κβάντιση και τη κωδικοποίηση εντροπίας καθώς και τις αντίστροφές τους διαδικασίες ένας κωδικοποιητής διαλέγει κατά τη διάρκεια της κωδικοποίησης ενός βίντεο μεταξύ δύο τεχνικών συμπίεσης, τη τεχνική Intra-frame και τη τεχνική Inter-frame.

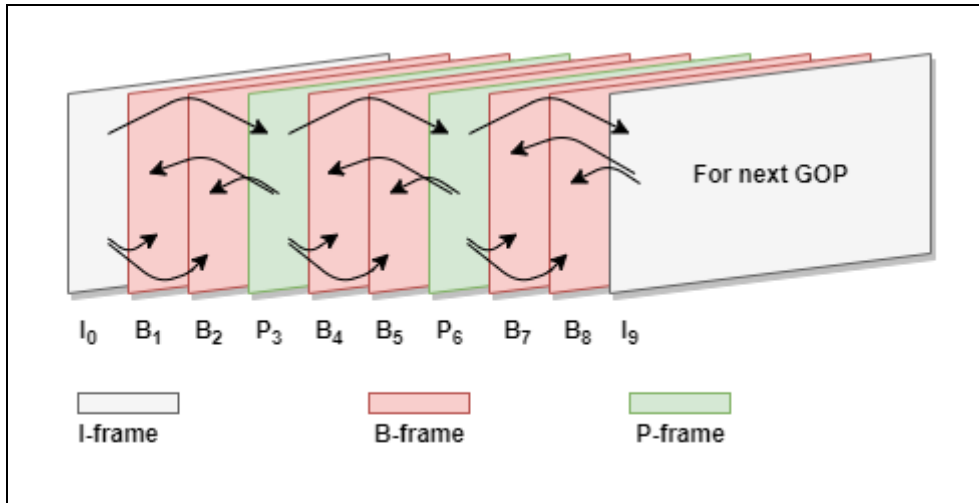
## Συμπίεση Intra και Inter-frame

---

Όπως είδαμε και στις προηγούμενες παραγράφους ένα αναπόσπαστο μέρος της κωδικοποίησης βίντεο είναι οι δύο τεχνικές συμπίεσης που μπορούν να εφαρμοστούν σε μία εικόνα, η τεχνική Intra-frame και η τεχνική Inter-frame. Πρόκειται για δύο τεχνικές πρόβλεψης (prediction) που εφαρμόζονται σε μία εικόνα για τη κωδικοποίηση της με σκοπό τη μείωση του μεγέθους τους, την αφαίρεση πλεονάζουσας πληροφορίας χωρίς να υπάρχει σημαντική απώλεια στη ποιότητα της εικόνας ή βίντεο. Στις δύο τεχνικές αυτές έχουν χωριστεί ένα σύνολο διαφορετικών αλγορίθμων που υλοποιούνται στο πρότυπο με τη διαφορά ότι εκμεταλλεύονται ένα διαφορετικό είδος πλεονασμού στην εικόνα.

Η πρόβλεψη Intra-frame (Intra-frame Prediction) αφορά τεχνικές που εφαρμόζονται εντός μίας εικόνας για να επιτευχθεί η συμπίεση της εικόνας και κατά συνέπεια ενός ολόκληρου βίντεο. Σε μία εικόνα, οι τιμές των εικονοστοιχείων της συνήθως έχουν μεγάλο βαθμό συσχέτισης μεταξύ τους όπως για παράδειγμα στη περίπτωση ενός τοίχου στο φόντο της εικόνας ή κάποιου άλλου μονότονου μέρους της οπότε η πρόβλεψη Intra-frame εκμεταλλεύεται το χωρικό πλεονασμό (spatial redundancy) της εικόνας εφαρμόζοντας τεχνικές που αφορούν το ίδιο το καρέ και τα εικονοστοιχεία του. Εφόσον η πρόβλεψη Intra-frame αφορά ένα καρέ τότε στη συμπίεση μπορούν να χρησιμοποιηθούν τεχνικές που εφαρμόζονται στη ψηφιακή επεξεργασία εικόνας. Επιπλέον, όσον αφορά την απόδοση της πρόβλεψης προσφέρει καλή συμπίεση χωρίς αρκετές απώλειες στην ποιότητα της εικόνας και επειδή χρησιμοποιεί μία εικόνα στην υλοποίηση απαιτεί μικρή επεξεργαστική ισχύ όπως και λίγο χρόνο.

Από την άλλη, η πρόβλεψη Inter-frame αφορά τεχνικές συμπίεσης που εφαρμόζονται σε ένα πλήθος συνεχόμενων εικόνων οι οποίες ονομάζονται Group of Pictures (GOP) που αποτελούνται από διαφορετικά είδη εικόνων και χωρίζονται βάσει του τρόπου συμπίεσής τους και τεχνικών που εφαρμόζονται σε αυτές. Αυτή η μέθοδος πρόβλεψης είναι ένας τρόπος κωδικοποίησης και αποθήκευσης των αλλαγών μεταξύ των εικόνων σε ένα βίντεο, εκμεταλλεύοντας το χρονικό πλεονασμό (temporal redundancy) που υπάρχει μεταξύ τους. Πιο συγκεκριμένα για το GOP, υπάρχουν τα I-frames δηλαδή καρέ τα οποία κωδικοποιούνται με συμπίεση Intra, τα P-frames δηλαδή τα Predicted καρέ που έχουν σαν αναφορά προηγούμενα καρέ που έχουν ήδη κωδικοποιηθεί και τέλος υπάρχουν τα B-frames δηλαδή τα Bi-directional predicted καρέ τα οποία χρησιμοποιούν σαν αναφορά προηγούμενα καθώς και επόμενα καρέ σαν αναφορά για τη κωδικοποίησή τους και προσφέρουν τη καλύτερο βαθμό συμπίεσης. Στην απόδοση της Inter-frame πρόβλεψης, προσφέρει καλύτερη βαθμό συμπίεσης σε σχέση με την Intra-frame όμως απαιτεί μεγαλύτερη επεξεργαστική ισχύ και περισσότερο χρόνο.



Εικόνα 5. GOP μήκους 8 με τις εξαρτήσεις.

## ΚΕΦΑΛΑΙΟ 2 Βιβλιογραφική Επισκόπηση

---

Σκοπός αυτής της πτυχιακής εργασίας είναι η μείωση της πολυπλοκότητας του νέου προτύπου συμπίεσης video VVC χρησιμοποιώντας το κριτήριο διακύμανσης με τον τελεστή ή το φίλτρο Sobel στην Intra-frame συμπίεση του αλγορίθμου. Αν και τυποποιήθηκε το νέο πρότυπο το 2020, πέρα από τις ήδη υπάρχουσες ερευνητικές εργασίες που εφαρμόζουν το φίλτρο Sobel στο προηγούμενο πρότυπο συμπίεση HEVC, έχουν δημοσιευθεί και εργασίες πάνω στο νεότερο πρότυπο χρησιμοποιώντας παρόμοιες τεχνικές. Σε αυτό το κεφάλαιο θα γίνει μία αναφορά σε μερικές ερευνητικές εργασίες στις οποίες χρησιμοποιήθηκε το φίλτρο Sobel με σκοπό την εύρεση ακμών μέσω των κλίσεων που προσφέρουν τα διαφορετικά φίλτρα.

Στο [15], ο Dong και άλλοι εφάρμοσαν ένα σύνολο αλγορίθμων με σκοπό τη μείωση χρόνου απόφασης για την επιλογή mode συμπίεσης στο VVC. Πιο συγκεκριμένα, χρησιμοποίησαν 9 χαρακτηριστικά για τη μείωση των συνολικών mode που ελέγχει ο κωδικοποιητής εκ των οποίων σε ένα από αυτά χρησιμοποίησαν το κάθετο και οριζόντιο φίλτρο Sobel για να αναπαραστήσουν την ισχύ των αλλαγών στην υφή των CU. Στο [18], ο Yang και άλλοι εφάρμοσαν το φίλτρα Sobel σε ένα σημείο της εργασίας τους με σκοπό τη συλλογή πληροφοριών για το πόσο ομοιογενές είναι η υφή του CU στο VVC. Ένα CU με χαμηλή τιμή ομοιογένειας δείχνει μία περιοχή που ανήκει στο παρασκήνιο του βίντεο ενώ μία μεγαλύτερη τιμή δείχνει ένα περίπλοκο σημείο στο βίντεο το οποίο θα χωριστεί περισσότερο. Στο [19], με παρόμοιο τρόπο ο Chen και άλλοι χρησιμοποιούν το τελεστή Sobel μαζί με μία κατώτατη τιμή (threshold value) σε ένα βήμα μίας διαδικασίας για την απόφαση του περαιτέρω διαχωρισμού ή όχι ενός 32x32 CU βάσει της ομοιογένειας του.

Όσον αφορά το προηγούμενο πρότυπο συμπίεσης HEVC, στην ερευνητική εργασία [16] τα φίλτρα Sobel χρησιμοποιήθηκαν παράλληλα με το μετασχηματισμό Hadamard για να συγκριθούν οι αποδόσεις τους με το τελεστή Prewitt σε μία διαδικασία εύρεσης προτεινόμενης κατεύθυνσης των εικονοστοιχείων για τη μείωση των συνολικών mode συμπίεσης. Τέλος, στο [17] ο Belghith και άλλοι εφάρμοσαν τα φίλτρα Sobel και έπειτα υπολόγισαν τη κλίση σε κάθε LCU με σκοπό την απόφαση τερματισμού του περαιτέρω διαχωρισμού του. Έχοντας υπολογίσει τη κλίση και ένα σύνολο κατώτατων τιμών για κάθε βίντεο, ο προτεινόμενος αλγόριθμος ελέγχει αν ένα LCU θεωρείται ότι έχει υφή και στη συνέχεια αποφασίζει σύμφωνα με τη κατώτατη τιμή αν θα χωριστεί από το πρότυπο.

## ΚΕΦΑΛΑΙΟ 3 Πρότυπο Συμπίεσης Video VVC

---

### Πρότυπο κωδικοποίησης βίντεο VVC/H.266

---

Με τις ολοένα και περισσότερο αυξανόμενες αναλύσεις στο σύνολο των συσκευών αναπαραγωγής βίντεο όπως τηλεοράσεις, οθόνες κινητών, οθόνες φορητών και σταθερών υπολογιστών και άλλα, καθώς και τους ταχύτερους ρυθμούς ανανέωσης στις προαναφερθείσες οθόνες και τέλος την χρήση ευρύτερων χρωματικών χώρων μέσω νέων τεχνολογιών όπως αυτή του Υψηλού Δυναμικού Εύρους (HDR) αυξήθηκε εκθετικά η πληροφορία που χρειάζεται να μεταδοθεί και να επεξεργαστεί. Με τις υψηλές αυτές αναλύσεις και το μεγαλύτερο βάθος bit να είναι διαθέσιμα σε συσκευές στην αγορά, εμφανίστηκε η ανάγκη για τη δημιουργία ενός νέου προτύπου συμπίεσης βίντεο που θα αντεπεξέρχεται στις καινούργιες ανάγκες προσφέροντας όχι μόνο καλύτερο ρυθμό συμπίεσης σε σχέση με προηγούμενα πρότυπα αλλά και τη δυνατότητα συμπίεσης μεγαλύτερων αναλύσεων, βάθος bit, ρυθμών ανανέωσης και άλλα τα οποία είτε δεν έχουν αναπτυχθεί είτε δεν χρησιμοποιούνται ακόμη σε συσκευές. Οπότε το 2017, η ομάδα JVET ξεκίνησε επίσημα τη διαδικασία τυποποίησης ενός νέου προτύπου που θα καλύπτει τις καινούργιες εμπορικές ανάγκες [13]. Τέλος, όπως αναφέρθηκε και στη προηγούμενη παράγραφο, η ομάδα JVET τυποποίησε τη πρώτη έκδοση του νέου προτύπου κωδικοποίησης βίντεο το 2020 με αριθμούς ITU H.266 και ISO/IEC 23090-3.

### Καινούργια χαρακτηριστικά στο πρότυπο VVC/H.266

---

Στο νέο πρότυπο κωδικοποίησης βίντεο VVC έχει προστεθεί ένα μεγάλο πλήθος νέων χαρακτηριστικών σε σχέση με το προηγούμενο πρότυπο HEVC/H.265 τα οποία από τη μία κάνουν πιο αποτελεσματική τη συμπίεση του βίντεο από την άλλη όμως υπάρχει το κόστος της αυξημένης πολυπλοκότητας στη κωδικοποίηση και την αποκωδικοποίηση του βίντεο. Σε αυτό το σημείο θα γίνει μία αναφορά σε κάποια από αυτά τα χαρακτηριστικά.

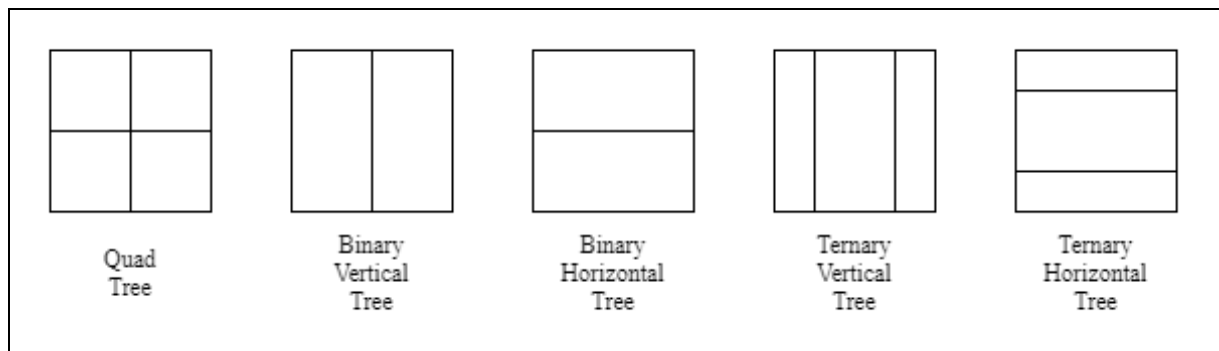
#### Εισαγωγή Δέντρου Πολλαπλών Τύπων και νέα μεγέθη CTU

---

Ένα από τα κύρια καινούργια χαρακτηριστικά του προτύπου είναι η εισαγωγή ενός νέου τρόπου διαμέρισης των μονάδων κωδικοποίησης ή Coding Unit (CU) στα δέντρα κωδικοποίησης μονάδων ή Coding Tree Unit (CTU). Στο προηγούμενο πρότυπο, ο τρόπος διαμέρισης ενός CTU γινόταν χωρίζοντας το σε 4 ίσα τετράγωνα μέρη μέσω ενός Τετραπλού Δέντρου (Quad Tree, QT), αποφασίζοντας έπειτα αν θα χωρίσει το κάθε CU με τον ίδιο τρόπο δημιουργώντας κατ' αυτόν το τρόπο ένα μεγάλο πλήθος από τετραγωνάκια. Στο νέο πρότυπο, έγινε η εισαγωγή του Δέντρου Πολλαπλών Τύπων (Multi Type Tree, MTT) το οποίο αποφασίζει, μέσω ενός κόστους, τη χρήση διαφορετικών τρόπων διαμέρισης των CTU.



Πιο συγκεκριμένα, ο κωδικοποιητής μπορεί να χωρίσει ένα CU χρησιμοποιώντας ένα Δυαδικό Δέντρο (Binary Tree, BT), το οποίο χωρίζει το CU σε δύο ίσα μέρη, ή ένα Τριαδικό Δέντρο (Ternary Tree, TT) το οποίο χωρίζει το CU σε τρία μέρη με την αναλογία στα μεγέθη που έχουν χωριστεί να είναι 1:2:1. Επιπλέον, ο κωδικοποιητής μπορεί να εφαρμόζει αυτά τα δέντρα χρησιμοποιώντας διαφορετική κλίση, δηλαδή διαμερίζοντας με αυτά οριζόντια και κάθετα, δίνοντας στο κωδικοποιητή ακόμη μεγαλύτερη ευελιξία στο τρόπο διαμέρισης. Επομένως, το νέο πρότυπο μπορεί να εφαρμόσει στο σύνολο 5 διαφορετικά δέντρα για τη κωδικοποίηση, το Τετραπλό Δέντρο, το Οριζόντιο Δυαδικό Δέντρο (BTH), το Κάθετο Δυαδικό Δέντρο (BTV), το Οριζόντιο Τριαδικό Δέντρο (TTH) και τέλος το Κάθετο Τριαδικό Δέντρο (TTV). Στη παρακάτω εικόνα απεικονίζονται τα διαφορετικά δέντρα.



Εικόνα 6. Οι 5 διαφορετικοί τρόποι χωρισμού ενός CU. Το Quad Tree και οι νέοι τρόποι του Δέντρου Πολλαπλών Τύπων.

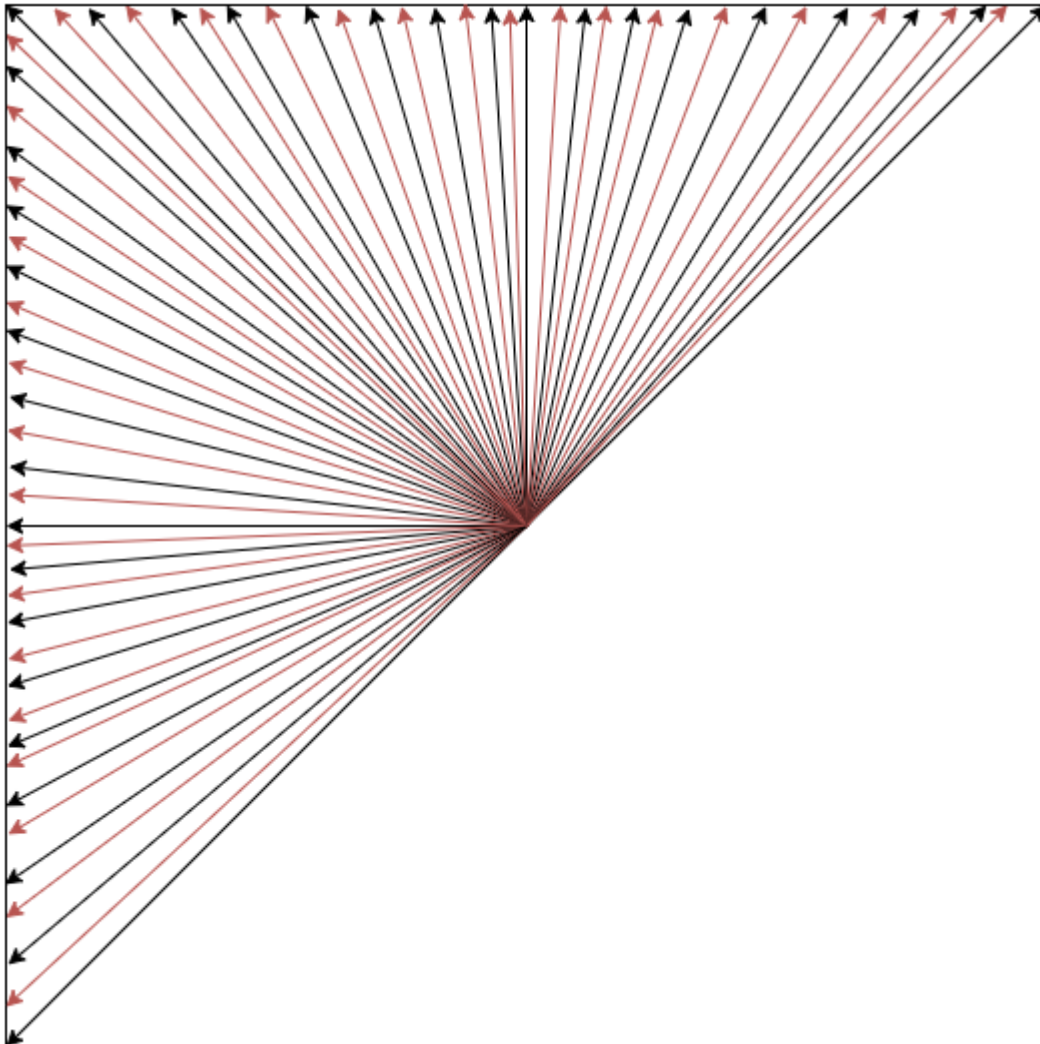
Επιπλέον, ένα νέο χαρακτηριστικό που προσφέρει το νέο πρότυπο είναι η αύξηση του μεγέθους των CTU που αφορούν το στοιχείο της φωτεινότητας (luma components) από 64x64 σε 128x128 και τα CTU που αφορούν το στοιχείο του χρώματος (chroma component) από 32x32 σε 64x64. Υπάρχει μία εξαίρεση σε αυτό καθώς τα μπλοκ μετασχηματισμού των αντίστοιχων στοιχείων πρέπει να έχουν μικρότερο μέγεθος από το μέγιστο μέγεθος που επιτρέπεται γενικά στο πρότυπο. Δηλαδή τα luma transform blocks πρέπει να έχουν μέγεθος 64x64 και τα chroma transform blocks να έχουν μέγεθος 32x32.

## Intra Prediction

### Αύξηση γωνιακών mode πρόβλεψης

Στο προηγούμενο πρότυπο συμπίεσης, στην Intra πρόβλεψη, ο κωδικοποιητής χρησιμοποιούσε συνολικά 35 mode πρόβλεψης κατεύθυνσης (directional intra modes), με τα πρώτα δύο να είναι το Planar, το DC και τα υπόλοιπα να είναι τα 33 γωνιακά (angular) modes. Στο νέο πρότυπο έχει γίνει η εισαγωγή 32 καινούργιων mode που αφορούν τη πρόβλεψη γωνίας εντός ενός CU έτσι ώστε να υπάρχει μεγαλύτερη ακρίβεια σε αυτό το είδος πρόβλεψης. Με αυτή τη προσθήκη, ο συνολικός αριθμός γωνιακών mode έχει αυξηθεί από τα 33 του προηγούμενου προτύπου συμπίεσης στα 65, φτάνοντας συνολικά τα 67 modes συμπίεσης Intra. Στον παρακάτω πίνακα φαίνεται πως έχουν διαμεριστεί τα νέα

modes στο πρότυπο. Πιο συγκεκριμένα, οι κατευθύνσεις με κόκκινο είναι οι 33 νέες γωνιακές modes και έχουν μπει ανάμεσα από τις κατευθύνσεις του προηγούμενου προτύπου.



**Εικόνα 7.** Οι 65 γωνιακές modes του νέου προτύπου, με τα μαύρα βελάκια να υποδεικνύουν τις 33 κατευθύνσεις του HEVC και τις υπόλοιπες 33 κόκκινες αυτές του VVC.

## Αύξηση Most Probable Modes

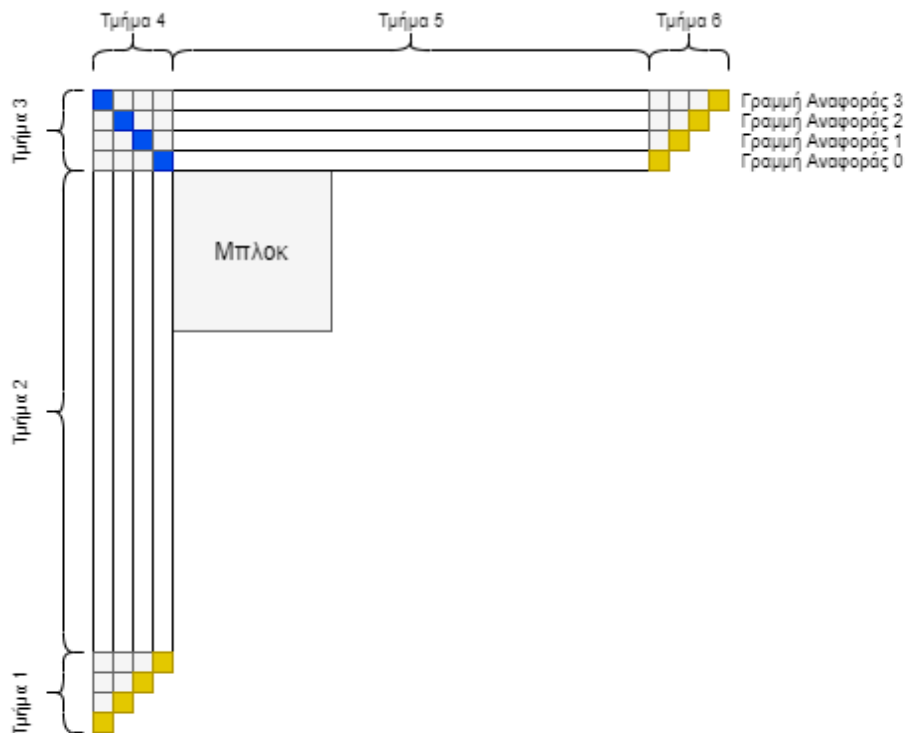
Επιπλέον, στη συμπίεση Intra τα πρότυπα υιοθετούν μία διαδικασία επιλογής των περισσότερο πιθανών mode συμπίεσης στα οποία θα υπολογιστεί το χρονοβόρο και υπολογιστικά πολύπλοκο κόστος RD για να γίνει η τελική επιλογή mode συμπίεσης. Η διαδικασία επιλογής των περισσότερο πιθανών mode ονομάζεται Most Probable Modes (MPM). Στο προηγούμενο πρότυπο HEVC/H.265 ο κωδικοποιητής διαλέγει 3 modes στα οποία θα υπολογιστεί το κόστος RD με σκοπό την επιλογή mode συμπίεσης, ενώ στο νέο πρότυπο συμπίεσης VVC αυτός ο αριθμός έχει αυξηθεί από 3 σε 6 δίνοντας στον κωδικοποιητή τη δυνατότητα να είναι περισσότερο ακριβής. Η διαδικασία επιλογής αυτής της λίστας λειτουργεί ελέγχοντας τα modes που επέλεξαν προηγούμενα Intra κωδικοποιημένα μπλοκ και

αναλόγως δημιουργεί την λίστα, ελέγχοντας συγκεκριμένα το αριστερό και τα πάνω μπλοκ από το τρέχον μπλοκ που κωδικοποιεί. Παρακάτω ακολουθεί η διαδικασία αυτή στο πρότυπο VVC.

- Αν δεν υπάρχει κάποιο γειτονικό μπλοκ, επιλέγεται η Planar mode για τη κωδικοποίηση χωρίς να δημιουργηθεί η λίστα
- Αν για το Αριστερό και το Πάνω μπλοκ έχει επιλεγεί μη γωνιακό mode:
  - Λίστα MPM -> {Planar, DC, Κάθετη, Οριζόντια, Κάθετη - 4, Κάθετη + 4}
- Αν μία από τις modes είναι γωνιακή και η άλλη μη γωνιακή:
  - Θέτει τη μεγαλύτερη mode σαν Μέγιστη:
  - Λίστα MPM -> {Planar, Μέγιστη, Μέγιστη - 1, Μέγιστη + 1, Μέγιστη - 2, Μέγιστη + 2}
- Αν έχουν επιλεγεί διαφορετικά γωνιακά modes:
  - Θέτει σαν Μέγιστη τη μεγαλύτερη mode από τα Αριστερό και Πάνω μπλοκ
  - Θέτει σαν Ελάχιστη τη μικρότερη mode από τα Αριστερό και Πάνω μπλοκ
  - Αν Μέγιστη - Ελάχιστη ισούται με 1:
    - Λίστα MPM -> {Planar, Αριστερή, Πάνω, Ελάχιστη - 1, Μέγιστη + 1, Ελάχιστη - 2}
    - Αλλιώς αν Μέγιστη - Ελάχιστη είναι μεγαλύτερο ή ίσο με 62:
      - Λίστα MPM -> {Planar, Αριστερή, Πάνω, Ελάχιστη + 1, Μέγιστη - 1, Ελάχιστη + 2}
    - Αλλιώς αν Μέγιστη - Ελάχιστη ισούται με 2:
      - Λίστα MPM -> {Planar, Αριστερή, Πάνω, Ελάχιστη + 1, Ελάχιστη - 1, Μέγιστη + 1}
    - Αλλιώς:
      - Λίστα MPM -> {Planar, Αριστερή, Πάνω, Ελάχιστη - 1, -Ελάχιστη + 1, Μέγιστη - 1}
  - Αν η Αριστερή και η Πάνω mode είναι η ίδια γωνιακή mode:
    - Λίστα MPM -> {Planar, Αριστερή, Αριστερή - 1, Αριστερή + 1, Αριστερή - 2, Αριστερή + 2}

### Multiple Reference Line (MRL)

Το προηγούμενο πρότυπο συμπίεσης βίντεο HEVC χρησιμοποιούσε μία γραμμή αναφοράς για να παράγει τη πρόβλεψη στην Intra-frame πρόβλεψη. Στο νέο πρότυπο έχει προστεθεί ένα νέο χαρακτηριστικό το οποίο επιτρέπει στον κωδικοποιητή να χρησιμοποιήσει πολλαπλές γραμμές αναφοράς για να επιτευχθεί η πρόβλεψη και ονομάζεται Multiple Reference Line (MRL). Ο κωδικοποιητής το εφαρμόζει στα μπλοκ ενός CTU με τον περιορισμό ότι δεν θα αναζητήσει γραμμές αναφοράς στη πρώτη σειρά μπλοκ κωδικοποίησης καθώς θα αναζητούσε γραμμές που ανήκουν σε προηγούμενα, ήδη κωδικοποιημένα CTU από το τρέχον. Στη παρακάτω εικόνα απεικονίζονται οι πολλαπλές γραμμές αναφοράς, με τη γραμμή αναφοράς 0 να είναι αυτή που επέλεγε το πρότυπο HEVC ενώ στο νέο πρότυπο κωδικοποίησης VVC αξιοποιούνται δύο επιπλέον γραμμές αναφοράς, η 1 και η 3.



Εικόνα 8. Πολλαπλές Γραμμές Αναφοράς (MRL),

Αυτά είναι κάποια από τα χαρακτηριστικά που έχουν προστεθεί στο νέο πρότυπο σε σχέση με το προηγούμενο, τα οποία βοηθάνε στη πιο αποτελεσματική συμπίεση ενός βίντεο κρατώντας την ίδια ποιότητα σε σχέση με το προηγούμενο πρότυπο αυξάνοντας όμως την υπολογιστική πολυπλοκότητα και το χρόνο κωδικοποίησης και αποκωδικοποίησης.

## Inter Prediction

### Εκτεταμένη πρόβλεψη συγχώνευσης

Στην Inter-frame πρόβλεψη υπάρχει η mode συγχώνευσης σε ένα CU όπου οι παράμετροι κίνησης λαμβάνονται από γειτονικά CUs. Στο πρότυπο VVC έχει επεκταθεί η πρόβλεψη συγχώνευσης καθώς έχει δημιουργηθεί μια λίστα υποψήφιων CUs για συγχώνευση. Αυτές οι παράμετροι ονομάζονται διανύσματα κίνησης (motion vectors) και η πρόβλεψη που χρησιμοποιεί αυτά τα διανύσματα ονομάζεται Motion Vector Predictors (MVPs).

1. Χωρικά MVP από τα γειτονικά CUs που αφορούν το χώρο
2. Χρονικά MVP από CUs στην ίδια θέση σε άλλη εικόνα
3. Ιστορικά βασισμένα MVP από ένα πίνακα FIFO
4. Κατά ζεύγη MVP
5. Μηδενικά MVP

## Χρονική πρόβλεψη διανύσματος κίνησης με βάση υπο-μπλοκ (SbTMVP)

Στο πρότυπο HEVC χρησιμοποιήθηκε η τεχνική πρόβλεψης TMVP, όπου για να εφαρμοστεί η πρόβλεψη στο τρέχον CU ο κωδικοποιητής συλλέγει πληροφορίες για τα mode πρόβλεψης, τα διανύσματα κίνησης της φωτεινότητας, δείκτες αναφοράς μεταξύ άλλων από το CU που βρίσκεται στο ίδιο σημείο από μία εικόνα αναφοράς. Στο VVC έχει προστεθεί ένα νέο χαρακτηριστικό που βασίζεται στο TMVP του HEVC όπου μπαίνει σε επίπεδο υπο-μπλοκ για μεγαλύτερη ακρίβεια και ονομάζεται Sub-block TMVP. Επιπλέον, στο SbTMVP εφαρμόζεται μία μετατόπιση κίνησης πριν ανακτήσει τις χρονικές πληροφορίες για τη κίνηση όπου η μετατόπιση κίνησης έχει υλοποιηθεί με τα χωρικά διανύσματα κίνησης από ένα από γειτονικά CUs.

## Προσαρμοσμένη ανάλυση διανυσμάτων κίνησης (AMVR)

Στο πρότυπο HEVC, οι διαφορές διανυσμάτων κίνησης ή motion vector differences (MVD), που είναι η διαφορά μεταξύ του διανύσματος κίνησης και του διανύσματος κίνησης του προβλεπόμενου CU, σηματοδοτούνται σε μονάδες δειγμάτων φωτεινότητας του  $\frac{1}{4}$  όταν η σημαία `use_integer_mv_flag` ισούται με 0 στην επικεφαλίδα του slice. Στο VVC, προστέθηκε μία επέκταση του προηγούμενου χαρακτηριστικού που ονομάζεται AMVR και εφαρμόζεται σε επίπεδο CU, επιτρέποντας στα MVD να κωδικοποιηθούν με διαφορετική ακρίβεια ανάλογα με το mode που θα επιλεγεί.

- Κανονική AMVP mode:  $\frac{1}{4}$  δείγματος φωτεινότητας, μισό δείγμα φωτεινότητας, ακέραιο δείγμα φωτεινότητας ή 4 δείγματα φωτεινότητας.
- Αφινικό AMVP mode:  $\frac{1}{4}$  δείγματος φωτεινότητας, ακέραιο δείγμα φωτεινότητας ή  $\frac{1}{16}$  δείγματος φωτεινότητας.

Η ένδειξη ανάλυσης διαφοράς διανυσμάτων κίνησης σε επίπεδο CU σηματοδοτείται υπό τον όρο ότι το τρέχων CU έχει τουλάχιστον ένα μη μηδενικό στοιχείο διαφοράς διανύσματος κίνησης. Αν όλα στοιχεία MVD είναι μηδενικά, συμπεραίνεται η χρήση της ανάλυσης MVD  $\frac{1}{4}$  δείγματος φωτεινότητας.

## ΚΕΦΑΛΑΙΟ 4 Ανίχνευση Ακμών

---

### Πληροφορίες για την ανίχνευση ακμών

---

Στη ψηφιακή επεξεργασία εικόνας υπάρχει ένα πλήθος αλγορίθμων και τεχνικών που εφαρμόζεται στις εικόνες με σκοπό την εξαγωγή πληροφοριών, συμπίεση εικόνας, αφαίρεση θορύβου και γενική επεξεργασία εικόνας όπως αντανάκλαση, κλιμάκωση και περιστροφή μίας εικόνας μέσω μετασχηματισμών μεταξύ άλλων. Στην εξαγωγή πληροφοριών από μία ψηφιακή εικόνα ανήκουν διάφορες τεχνικές όπως αυτές για την ανίχνευση σημείων, γραμμών και ακμών με την ανίχνευση ακμών να είναι αυτή που χρησιμοποιήσαμε σε αυτή τη πτυχιακή για την εξαγωγή δεδομένων με σκοπό τη μείωση των συνολικών γωνιακών mode. Με την ανίχνευση ακμών αναφερόμαστε στη διαδικασία όπου διάφορες μέθοδοι εφαρμόζονται στις εικόνες έχοντας ως στόχο την εύρεση και αναγνώριση ακμών στα αντικείμενα που εμφανίζονται στις εικόνες και έχει εφαρμογές στην ανίχνευση και εξαγωγή χαρακτηριστικών.

Η ανίχνευση ακμών λειτουργεί με διάφορους τρόπους με έναν από αυτούς να πραγματοποιείται με την εφαρμογή φίλτρων (filters) ή πυρήνων (kernels) στην αρχική εικόνα χρησιμοποιώντας τη διαδικασία του χωρικού φιλτραρίσματος (spatial filtering). Στο χωρικό φιλτράρισμα το φίλτρο εφαρμόζεται σε ολόκληρη την εικόνα αθροίζοντας τα στοιχεία του με τα αντίστοιχα στοιχεία της αρχικής εικόνας και έπειτα διαιρώντας το άθροισμα με το σύνολο των στοιχείων του φίλτρου. Στο τέλος αυτής της διαδικασίας έχουμε μία εικόνα με ένα σύνολο διαφορετικών τιμών. Όσο μεγαλύτερη είναι αυτή η τιμή τόσο πιο έντονη είναι η ακμή σε εκείνο το σημείο ενώ μία μικρότερη τιμή δείχνει την απουσία ακμής ή και την ύπαρξη μίας διαφορετικής ακμής.

Δύο γνωστοί ανιχνευτές ακμών είναι ο τελεστής Prewitt και ο τελεστής Sobel. Οι δύο αυτοί τελεστές χρησιμοποιούν δύο φίλτρα για την ανίχνευση ακμών, ένα για τον υπολογισμό της οριζόντιας παραγώγου που βοηθάει στην εύρεση κάθετων γραμμών και ένα για τον υπολογισμό της κάθετης παραγώγου που βοηθάει στην εύρεση οριζόντιων ακμών. Εφόσον έχουν εφαρμοστεί τα δύο φίλτρα σε μία εικόνα, θα υπάρχουν και δύο εικόνες με τις αντίστοιχες ακμές. Αυτές οι δύο εικόνες προστίθενται για να δημιουργηθεί και μία ολοκληρωμένη εικόνα με όλες τις ακμές που κατάφεραν και ανίχνευσαν τα δύο φίλτρα.

Ο τελεστής Prewitt είναι ένας ανιχνευτής ακμών ο οποίος υπολογίζει κατά προσέγγιση τη κλίση σε κάθε σημείο μίας εικόνας χρησιμοποιώντας δύο φίλτρα. Ένα για την οριζόντια παράγωγο και ένα για τη κάθετη παράγωγο. Τα φίλτρα φαίνονται στη παρακάτω εικόνα:

$$G_x = \begin{bmatrix} +1 & 0 & -1 \\ +1 & 0 & -1 \\ +1 & 0 & -1 \end{bmatrix} \quad G_y = \begin{bmatrix} +1 & +1 & +1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix}$$

Εικόνα 9. Οριζόντιο και κάθετο φίλτρο Prewitt.

Σε αυτή τη πτυχιακή εργασία διαλέξαμε το τελεστή Sobel για τη προσέγγιση της κλίσης σε κάθε CU που ελέγχει το πρότυπο VVC/H.266 έχοντας ως μέλημα τη μείωση των συνολικών mode συμπίεσης και κατ' επέκταση τη μείωση της πολυπλοκότητας.

## Τελεστής Sobel

Ο τελεστής Sobel ή φίλτρο Sobel είναι ένας πυρήνας συνέλιξης μεγέθους 3x3 ο οποίος εφαρμόζεται πάνω σε μία αρχική εικόνα και χρησιμοποιείται στην ψηφιακή επεξεργασία εικόνας για ανίχνευση ακμών. Όπως και ο τελεστής Prewitt, ο Sobel είναι ένας τελεστής διακριτής ή πεπερασμένης διαφοράς και υπολογίζει μία προσέγγιση της κλίσης μέσω των διαφορών στη φωτεινότητα που έχουν τα εικονοστοιχεία. Ο τελεστής Sobel έχει και αυτός δύο φίλτρα, ένα για την οριζόντια παράγωγο και ένα για τη κάθετη παράγωγο, παρόμοια με αυτά του τελεστή Prewitt, αλλά σε αντίθεση με το τελευταίο δίνει έμφαση στο κεντρικό στοιχείο χρησιμοποιώντας τον αριθμό 2. Ο τελεστής Sobel φέρει καλύτερα αποτελέσματα στην ανίχνευση ακμών και προτιμάτε από τους ερευνητές σε σύγκριση με το τελεστή Prewitt.

$$G_x = \begin{bmatrix} -1 & 0 & +1 \\ -2 & 0 & +2 \\ -1 & 0 & +1 \end{bmatrix} \quad G_y = \begin{bmatrix} +1 & +2 & +1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$

Εικόνα 10. Οριζόντιο και κάθετο φίλτρο Sobel.

Στον αλγόριθμο που υλοποιήθηκε για αυτή τη πτυχιακή εργασία με σκοπό τη μείωση της πολυπλοκότητας χρησιμοποιήθηκαν 4 πυρήνες Sobel, οι δύο συμβατικοί πυρήνες για την οριζόντια και κάθετη παράγωγο καθώς και δύο διαγώνιοι πυρήνες για τις γωνίες των 45 και 135 μοιρών. Τα φίλτρα αυτά έχουν την εξής μορφή:

$$G_x = \begin{bmatrix} -1 & 0 & +1 \\ -2 & 0 & +2 \\ -1 & 0 & +1 \end{bmatrix}$$

$$G_y = \begin{bmatrix} +1 & +2 & +1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$

$$G_{45} = \begin{bmatrix} 0 & +1 & +2 \\ -1 & 0 & +1 \\ -2 & -1 & 0 \end{bmatrix}$$

$$G_{135} = \begin{bmatrix} -2 & -1 & 0 \\ -1 & 0 & +1 \\ 0 & +1 & +2 \end{bmatrix}$$

Εικόνα 11. Φίλτρα Sobel που χρησιμοποιήθηκαν στον αλγόριθμο.



## ΚΕΦΑΛΑΙΟ 5 VVC Test Model

---

### Εισαγωγικό σημείωμα

---

Σε αυτό το κεφάλαιο θα γίνει μια αναφορά στο VVC Test Model (VTM), εξηγώντας πως εκτελείται και θα περιγραφεί ο τρόπος λειτουργίας της κωδικοποίησης στο VTM δίνοντας έμφαση στη συμπίεση Intra-frame. Έπειτα θα αναλυθεί πλήρως ο αλγόριθμος για τη μείωση της πολυπλοκότητας στο νέο πρότυπο VVC χρησιμοποιώντας τα φίλτρα Sobel.

### VTM, εκτέλεση και ροή του κωδικοποιητή

---

Το VTM είναι ανεπτυγμένο από την ομάδα JVET και χρησιμοποιείται για τη "μελέτη και υλοποίηση" [1] του κωδικοποιητή και αποκωδικοποιητή του νέου προτύπου από διάφορους χρήστες και είναι γραμμένο στη γλώσσα προγραμματισμού C++. Πρόκειται για το λογισμικό αναφοράς για τη σύσταση της ένωσης ITU-T H.266 και το πρότυπο του οργανισμού ISO σε συνεργασία με την επιτροπή IEC ISO/IEC 23090-3. [1][2][3] Το λογισμικό βρίσκεται στον ιστότοπο <[https://vcgit.hhi.fraunhofer.de/jvet/VVCSoftware\\_VTM](https://vcgit.hhi.fraunhofer.de/jvet/VVCSoftware_VTM)> μαζί με οδηγίες εγκατάστασης σε διάφορα λογισμικά σε συνδυασμό με ένα Ολοκληρωμένο περιβάλλον ανάπτυξης με τη χρήση του εργαλείου ανοιχτού κώδικα CMake.

Όσον αφορά την εκτέλεση του, διαλέγοντας σαν startup project το EncoderApp και εισάγοντας σαν command line argument στο Debugging Properties το -c config.cfg, όπου config ένα αρχείο τύπου .cfg, εκτελείται ο κωδικοποιητής του λογισμικού VTM με είσοδο οποιοδήποτε .yuv αρχείο. Με παρόμοια διαδικασία και εισάγοντας σαν command line argument το DecoderApp -b str.bin -o dec.yuv εκτελείται ο αποκωδικοποιητής του λογισμικού VTM έχοντας ως είσοδο το παραγόμενο από τη προηγούμενη διαδικασία αρχείο bitstream και σαν έξοδο το τελικό αποκωδικοποιημένο αρχείο.

Ο κωδικοποιητής στο VTM ονομάζεται EncoderApp και είναι ένα από τα projects στο λογισμικό αναφοράς. Αρχικά, όπως όλα τα προγράμματα στη C++, ο κωδικοποιητής ξεκινάει από τη συνάρτηση main() που βρίσκεται στο αρχείο encmain.cpp, εκεί υπάρχει ένα loop το οποίο κωδικοποιεί τα GOPs του sequence καλώντας τη συνάρτηση encode() του αρχείου EncApp.cpp. Έπειτα, στο EncApp.cpp καλεί μία άλλη συνάρτηση με το ίδιο όνομα, την encode() του αρχείου EncLib.cpp. Σε αυτή τη συνάρτηση θα καλέσει ο κωδικοποιητής τη compressGOP() του αρχείου EncGOP.cpp όπου με αυτή τη συνάρτηση το λογισμικό θα αρχίσει να χωρίζει ένα βίντεο από GOP σε slices, από slices σε ένα καρέ χωρισμένο σε CTUs, από τα CTUs προχωράει στη κωδικοποίηση ενός CTU και καταλήγει στο διαχωρισμό ενός CTU με το βέλτιστο τρόπο χρησιμοποιώντας το Multi Type Tree. Επομένως, στη compressGOP() καλείται η συνάρτηση compressSlice() του EncSlice.cpp για τη διαφορετική κωδικοποίηση ενός καρέ αναλόγως με το αν είναι I, P ή B slice. Προς το τέλος της compressSlice() υπάρχει η συνάρτηση encodeCtus() που ανήκει στο

ίδιο αρχείο, δηλαδή το EncSlice.cpp. Μέσα στη compressSlice() υπάρχει ένα for-loop όπου γίνεται η κωδικοποίηση του κάθε CTU χρησιμοποιώντας τη συνάρτηση compressCtu() του EncCu.cpp. Αξίζει να σημειωθεί ότι ένα καρέ μεγέθους 1920x1080 δηλαδή που έχει διαστάσεις Full HD χωρίζεται συνολικά σε 135 CTUs.

Από τη compressCtu() και έπειτα, ξεκινάνε διάφορες αναδρομικές συναρτήσεις με σκοπό την εύρεση του βέλτιστου τρόπου διαχωρισμού των CU που βρίσκονται στο κάθε CTU, αρχικά για το στοιχείο Y ενός βίντεο και έπειτα για τα δύο στοιχεία για τα χρώματά του, το Cb και το Cr όπου αντιστοιχούν στη μπλε και στη κόκκινη διαφορά. Την όλη διαδικασία τη ξεκινάει η συνάρτηση xCompressCU() του αρχείου EncCu.cpp για το στοιχείο της φωτεινότητας και στη συνέχεια σε δεύτερη κλήση της συνάρτησης καλείται για τα χρωματικά στοιχεία. Μέσα στη xCompressCU() υπάρχει ένα do-while loop που περιέχει όλα τα διαφορετικά είδη συμπίεσης ή modes όπως η συμπίεση Intra-frame και η Inter-frame. Εκτός από αυτά τα modes υπάρχει και η συνάρτηση xCheckModeSplit() στην οποία γίνεται ο διαχωρισμός των CU διαλέγοντας ένα από τα split ή τους τρόπους διαχωρισμού που προσφέρει το VVC, με το πρώτο split να είναι ένα Quad Split. Στη συνάρτηση xCheckModeSplit() υπάρχει ένα άλλο do-while loop όπου καλείται και πάλι η xCompressCU() για να ξεκινήσει η αναδρομή χωρίζοντας το CU σε μικρότερα. Εφόσον έχει τελειώσει ο χωρισμός του πρώτου CU η διαδικασία προχωράει στο επόμενο CU από το αρχικό Quad Split. Σε αυτή τη διαδικασία το λογισμικό εξαντλεί όλους του δυνατούς τρόπους χωρισμού ελέγχοντας το κόστος Rate Distortion (RD) και στο τέλος επιλέγει αυτό με το χαμηλότερο κόστος. Κατ' αυτό το τρόπο έχει χωρίσει το κάθε CU και κατά συνέπεια το κάθε CTU με το βέλτιστο τρόπο. Αυτό το επιτυγχάνει χρησιμοποιώντας δύο μεταβλητές, τη tempCS και τη bestCS που κρατάνε, μεταξύ άλλων, το τρόπο χωρισμού και το κόστος, και τη συνάρτηση xCheckBestMode() του αρχείου EncCu.cpp που επιλέγει μεταξύ των δύο.

Μέσω αυτής της διαδικασίας το λογισμικό αναφοράς VTM καταλήγει στο να έχει ένα CTU χωρισμένο με το βέλτιστο τρόπο και έπειτα ένα καρέ με διαχωρισμένα CTUs έτοιμα για κωδικοποίηση και τέλος μέσα από τα αρχικά loops προχωρώντας στα επόμενα frames καταλήγει στο να έχει ένα ολόκληρο κωδικοποιημένο sequence. Στη πτυχιακή αυτή οι αλλαγές αφορούν τη συμπίεση Intra-frame οπότε στο επόμενο κομμάτι θα αναλυθεί η ροή που ακολουθεί αυτή η μορφή συμπίεσης στο VTM καθώς και ο τρόπος λειτουργίας της.

## Ροή και τρόπος λειτουργίας συμπίεσης Intra-frame στο VTM

Εφόσον η συμπίεση είναι Intra-frame το VTM θα έχει επιλέξει τα I-Slices που αφορούν την συμπίεση Intra. Όπως και προηγουμένως καλείται πρώτα η συνάρτηση xCompressCU() για το στοιχείο Y του βίντεο και στη συνέχεια καλείται η xCheckModeSplit() για να ξεκινήσει ο διαχωρισμός του CTU. Στην xCheckModeSplit() θα κληθεί η xCompressCU() για την αναδρομική διαδικασία διαχωρισμού των CU. Στη συμπίεση Intra-frame, μέσα στο do-while loop της xCompressCU() θα πραγματοποιηθεί κλήση της xCheckRDCostIntra() του αρχείου EncCu.cpp, η οποία περιέχει τις συναρτήσεις estIntraPredLumaQT() και estIntraPredChromaQT() του αρχείου IntraSearch.cpp. Από τις δύο προηγούμενες

συναρτήσεις μας ενδιαφέρει η πρώτη, η `estIntraPredLumaQT()` όπου γίνεται η επιλογή του `mode` συμπίεσης από το σύνολο των 67 διαφορετικών `modes` για το κάθε CU.

Το κάθε CU περνάει από μια διαδικασία επιλογής μιας από τις 67 Intra `modes` του προτύπου VVC αφού έχει πρώτα δημιουργήσει μια λίστα υποψηφίων `modes`. Η διαδικασία για την επιλογή `modes` για εισαγωγή στη λίστα υποψηφίων `modes` για πλήρη έλεγχο στο κόστος Rate Distortion στη συμπίεση Intra στο VTM είναι χωρισμένη σε 3 σημεία και είναι η εξής:

- Αρχικά ελέγχει τα 33 συμβατικά (conventional) γωνιακά `modes` από το προηγούμενο πρότυπο κωδικοποίησης HEVC καθώς και τα `modes` Planar και DC.
- Έπειτα, ελέγχει τα εκτεταμένα (extended) γωνιακά `modes` που ανήκουν στο νέο πρότυπο VVC, ελέγχοντας τα δύο γειτονικά γωνιακά `modes`, με εξαίρεση τα `modes` Planar και DC τα οποία δεν είναι γωνιακά.
- Τέλος, ελέγχει τα 6 Most Probable Modes (MPM).

Στο τέλος αυτής της διαδικασίας έχει δημιουργηθεί μια λίστα υποψηφίων `modes` για πλήρη έλεγχο κόστους RD όπου έπειτα γίνεται και η τελική επιλογή. Αυτή η διαδικασία επαναλαμβάνεται για κάθε CU στους διαφορετικούς τρόπους κοψίματος τους σε όλα τα CTU στο τρέχον καρέ και για όσα καρέ πρόκειται να κωδικοποιηθούν.

## Δομή του αλγορίθμου

---

Ο προτεινόμενος αλγόριθμος αφορά την συμπίεση Intra-frame του κωδικοποιητή του νέου προτύπου συμπίεσης βίντεο VVC και χρησιμοποιεί φίλτρα Sobel σε κάθε CU κατά τη διάρκεια εύρεσης του βέλτιστου τρόπου χωρισμού των επί μέρους CTU μέσω μιας συνάρτησης. Το κάθε φίλτρο Sobel που χρησιμοποιείται αντιστοιχεί σε ένα υποσύνολο των 65 γωνιακών `modes` του προτύπου VVC. Εφόσον έχουν εφαρμοστεί τα φίλτρα Sobel, εκτελείται μία ταξινόμηση βάσει του μεγαλύτερου μέσου όρου τιμής εικονοστοιχείων σε ένα CU και διαλέγετε αυτό με το μεγαλύτερο μέσο όρο. Έπειτα, επιλέγεται το υποσύνολο των γωνιακών `modes` που αντιστοιχούν στο φίλτρο που επιλέχθηκε. Τέλος, με ένα περιορισμό επιλογής των `modes` σε δύο σημεία στο λογισμικό VTM περιορίζονται τα `modes` που ελέγχονται για κόστος Rate Distortion και επομένως ως επιλογή από τον ίδιο τον κωδικοποιητή. Ο αλγόριθμος εξηγείται πλήρως στις παρακάτω παραγράφους.

## Ο αλγόριθμος

---

Ο αλγόριθμος εφαρμόζεται μέσω μιας συνάρτησης με όνομα `sobelAnglePruning()` και ανήκει στις συναρτήσεις του αρχείου `IntraSearch()` και καλείται πριν το

λογισμικό αρχίσει τη διαδικασία επιλογής modes για την εισαγωγή στη λίστα υποψηφίων modes στο τρέχον CU. Στον αλγόριθμο χρησιμοποιούνται 4 structs στα οποία αντιστοιχεί μία από τις γωνίες των 4 φίλτρων Sobel που χρησιμοποιούνται στον αλγόριθμο με όνομα sobel\_x, sobel\_y, sobel\_45 και sobel\_135 για τα αντίστοιχα φίλτρα. Στα structs περιέχονται δύο μεταβλητές, η μία είναι η γωνία, με όνομα dirAngle, και η άλλη είναι αρχικά ένας μετρητής στον οποίο εν τέλει θα αποθηκευτεί ο μέσος όρος τιμής των εικονοστοιχείων έπειτα από την εφαρμογή του πυρήνα Sobel, με όνομα Average. Οπότε, αρχικά, στο τρέχον CU εφαρμόζονται τα τέσσερα φίλτρα Sobel περνώντας πάνω από όσα εικονοστοιχεία επιτρέπει το μέγεθος του φίλτρου αυξάνοντας το μετρητή για το μέσο όρο.

Εφόσον έχουν εφαρμοστεί τα τέσσερα φίλτρα υπολογίζεται ο μέσος όρος των τιμών από το κάθε φίλτρο Sobel στο CU στο οποίο βρίσκεται. Ο μεγαλύτερος μέσος όρος υποδεικνύει και τη μεγαλύτερη διακριτή διαφορά, δηλαδή κλίση, στο CU. Επιλέγεται, επομένως, η κλίση η οποία έχει και το μεγαλύτερο μέσο όρο έπειτα από την εφαρμογή του πυρήνα Sobel.

Τα 4 φίλτρα Sobel που χρησιμοποιούνται έχουν χωριστεί έτσι ώστε να αντιστοιχούν σε ένα υποσύνολο από τα 65 γωνιακά modes του νέου προτύπου κωδικοποίησης βίντεο VVC και έχει γίνει σε αυτή τη πτυχιακή με δύο διαφορετικούς τρόπους τους οποίους έχουμε ονομάσει Prop1 και Prop2 . Πιο συγκεκριμένα για το Prop1, στο φίλτρο για την οριζόντια παράγωγο των 0 μοιρών αντιστοιχούν τα modes από 35 έως 50, στο φίλτρο για τη διαγώνια παράγωγο των 135 μοιρών αντιστοιχούν τα modes από 51 έως 66, στο φίλτρο για την κάθετη παράγωγο των 90 μοιρών αντιστοιχούν τα modes από 2 έως 18 και τέλος στο φίλτρο για την άλλη διαγώνια παράγωγο των 45 μοιρών αντιστοιχούν τα modes από 19 έως 34. Ενώ για το Prop2 πολύ απλά έχουν ανταλλάξει θέση τα φίλτρα για την οριζόντια παράγωγο με τη κάθετη όπως και τα δύο διαγώνια μεταξύ τους δηλαδή η παράγωγος των 45 μοιρών με αυτή των 135. Αξίζει να σημειωθεί πως η αρίθμηση έχει γίνει όπως ισχύει και στο λογισμικό αναφοράς VTM άρα και στο πρότυπο, με τα modes 0 και 1 να αντιστοιχούν στους τρόπους συμπίεσης Planar και DC αντίστοιχα. Οπότε έχουμε:

Για το Prop1:

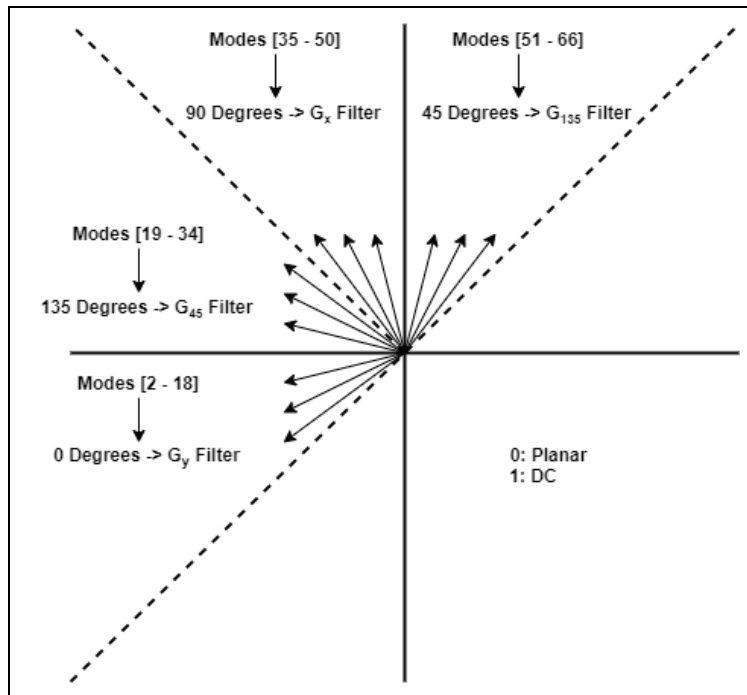
- Sobel\_y -> Modes 2 – 18
- Sobel\_45 -> Modes 19 – 34
- Sobel\_x -> Modes 35 – 50
- Sobel\_135 -> Modes 51 – 66

Ενώ για το Prop2:

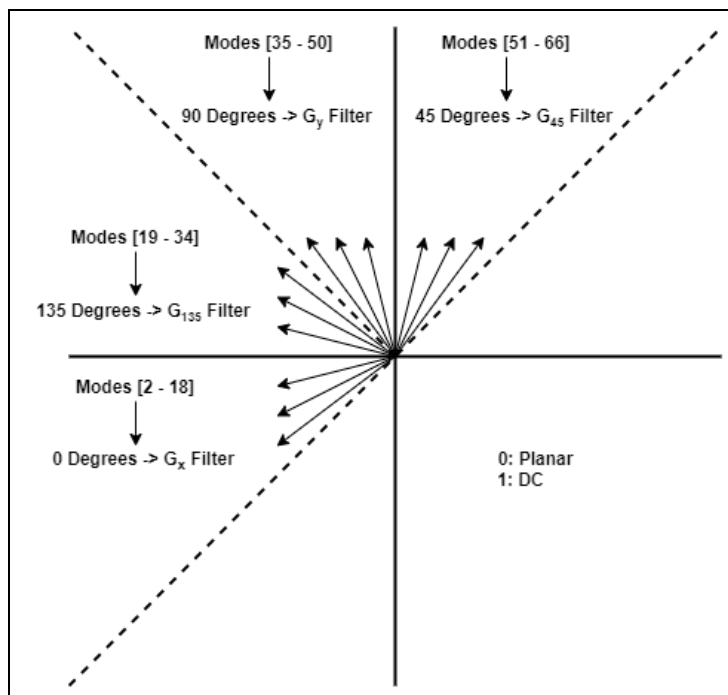
- Sobel\_x -> Modes 2 – 18
- Sobel\_135 -> Modes 19 – 34
- Sobel\_y -> Modes 35 – 50
- Sobel\_45 -> Modes 51 – 66

Στις παρακάτω εικόνες φαίνεται ο τρόπος χωρισμού των φίλτρων σε σχέση με το καρτεσιανό επίπεδο που χρησιμοποιείται για την απεικόνιση των γωνιακών modes.

Τα βελάκια χρησιμοποιούνται για να συμβολίσουν τα γωνιακά modes και είναι καθαρά ενδεικτικά, δεν αντιπροσωπεύουν δηλαδή το σύνολο των modes:



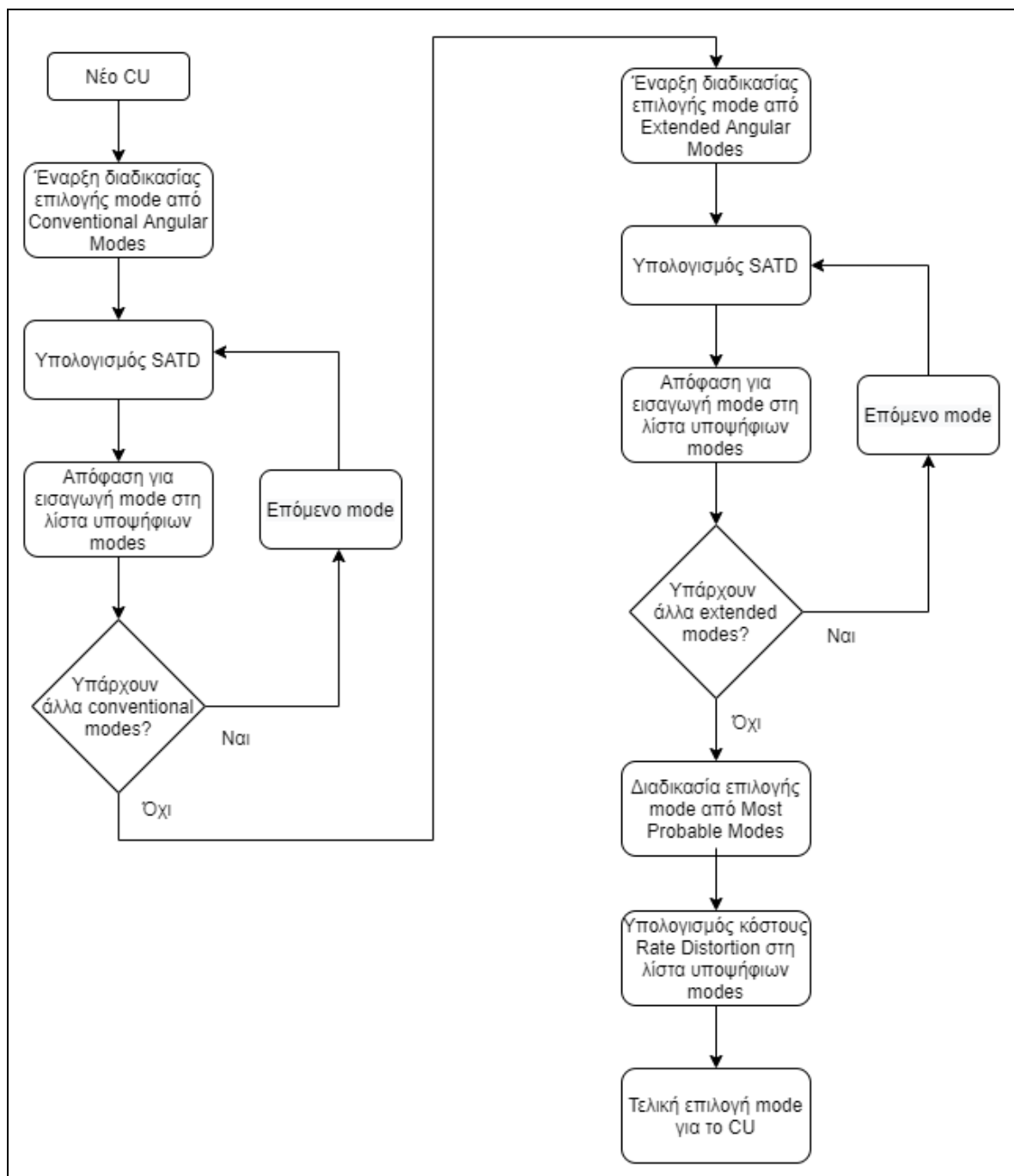
Εικόνα 12. Τρόπος χωρισμού Prop1 φίλτρων Sobel.



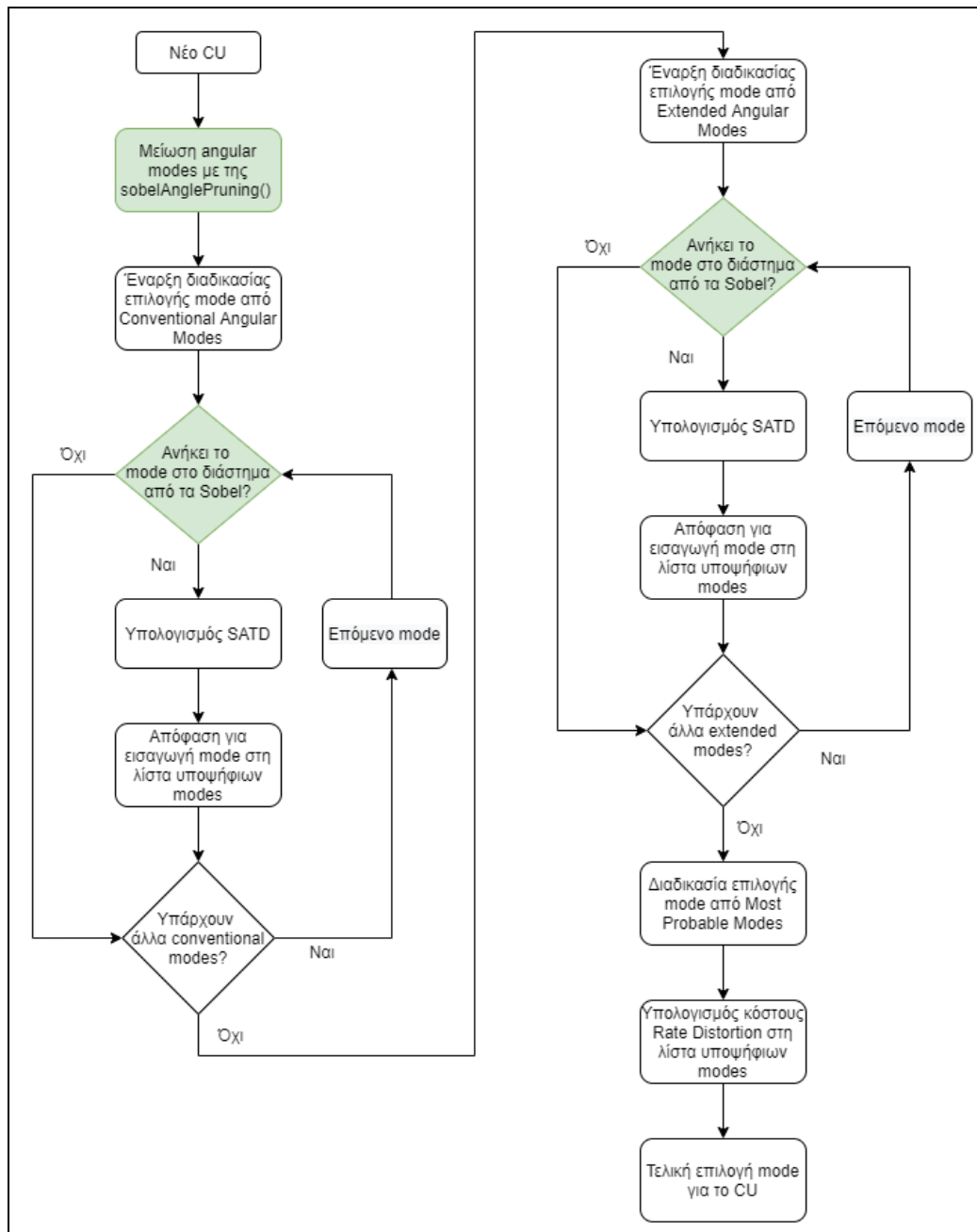
Εικόνα 13. Τρόπος χωρισμού Prop2 φίλτρων Sobel.

Εφόσον έχουν βρεθεί τα γωνιακά modes από το πυρήνα με το μεγαλύτερο μέσο όρο, μέσω δύο μεταβλητών για το αρχικό και τελικό mode, με ονόματα startMode και finalMode αντίστοιχα, και μιας δήλωσης if σε δύο μέρη του κώδικα περιορίζεται ο αριθμός των modes που ελέγχει το VTM στη συμπίεση Intra ελέγχοντας μόνο τα

modes που βρίσκονται μεταξύ των δύο μεταβλητών. Τα δύο μέρη στα οποία υπάρχει ο κώδικας με τη δήλωση if είναι στους δύο πρώτους ελέγχους που εκτελεί κατά τη διάρκεια εύρεσης modes για τη λίστα υποψήφιων modes, δηλαδή στους ελέγχους για τα συμβατικά και τα εκτεταμένα γωνιακά modes. Επιπλέον με τους περιορισμούς επιτρέπουμε τα modes συμπίεσης Planar και DC να ελέγχουν καθώς αυτά δεν αποτελούν μέρος των γωνιακών mode. Παρακάτω ακολουθούν δύο διαγράμματα ροής για τη πλήρη κατανόηση της ροής του αλγορίθμου επιλογής mode καθώς και τις αλλαγές που πραγματοποιήσαμε σε αυτόν όπως επίσης και δύο εικόνες με το ψευδοκώδικα και τις αλλαγές μεταξύ των δύο διαφορετικών τρόπων χωρισμού των φίλτρων.



Εικόνα 14. Διάγραμμα ροής αλγορίθμου επιλογής mode.



Εικόνα 15. Διάγραμμα ροής αλγορίθμου επιλογής mode με τις αλλαγές για τη μείωση συνολικών angular mode.

---

**Pseudocode for sobelAnglePruning() Prop1**

---

**sobelAnglePruning (CurrentCU, StartMode, FinalMode)**

```
{
    edgeDir[0].dirAngle = 0;
    edgeDir[1].dirAngle = 90;
    edgeDir[2].dirAngle = 45;
    edgeDir[3].dirAngle = 135;

    // Calculate sobel averages
    for iRows from 0 to cuHeight
    for iCols from 0 to cuWidth
        d1 = d2 = 0;
        for u from iRows - 1 to iRows + 2
            d2 = 0;
            for v from iCols - 1 to iCols + 2
                if Current pixels x and y within Picture
                    edgeDir[0].Average += ComponentY(x, y) * sobel_y[d1][d2];
                    edgeDir[1].Average += ComponentY(x, y) * sobel_x[d1][d2];
                    edgeDir[2].Average += ComponentY(x, y) * sobel_135[d1][d2];
                    edgeDir[3].Average += ComponentY(x, y) * sobel_45[d1][d2];
                end if
                d2++;
            d1++;

    edgeDir[0].Average = edgeDir[0].Average / (cuHeight * cuWidth);
    edgeDir[1].Average = edgeDir[1].Average / (cuHeight * cuWidth);
    edgeDir[2].Average = edgeDir[2].Average / (cuHeight * cuWidth);
    edgeDir[3].Average = edgeDir[3].Average / (cuHeight * cuWidth);

    // find max average Sobel value
    for i from 0 to 3
        for j from 0 to 3 - i
            if edgeDir[j].Average < edgeDir[j + 1].Average
                tempEdgeDir = edgeDir[j];
                edgeDir[j] = edgeDir[j + 1];
                edgeDir[j + 1] = tempEdgeDir;
            end if

    // in edgeDir[0] max average value from Sobel filters
    if edgeDir[0].dirAngle == 0
        StartMode = 2;
        FinalMode = 18;
    if edgeDir[0].dirAngle == 135
        StartMode = 19;
        FinalMode = 34;
    if edgeDir[0].dirAngle == 90
        StartMode = 35;
        FinalMode = 50;
    if edgeDir[0].dirAngle == 45
        StartMode = 51;
        FinalMode = 66;
}
```

---

Εικόνα 16. Ψευδοκώδικας για τον αλγόριθμο Prop1.



---

**Pseudocode for sobelAnglePruning() Prop2**

---

**sobelAnglePruning (CurrentCU, StartMode, FinalMode)**

```
{
    edgeDir[0].dirAngle = 0;
    edgeDir[1].dirAngle = 90;
    edgeDir[2].dirAngle = 45;
    edgeDir[3].dirAngle = 135;

    // Calculate sobel averages
    for iRows from 0 to cuHeight
        for iCols from 0 to cuWidth
            d1 = d2 = 0;
            for u from iRows - 1 to iRows + 2
                d2 = 0;
                for v from iCols - 1 to iCols + 2
                    if Current pixels x and y within Picture
                        edgeDir[0].Average += ComponentY(x, y) * sobel_x[d1][d2];
                        edgeDir[1].Average += ComponentY(x, y) * sobel_y[d1][d2];
                        edgeDir[2].Average += ComponentY(x, y) * sobel_45[d1][d2];
                        edgeDir[3].Average += ComponentY(x, y) * sobel_135[d1][d2];
                    end if
                    d2++;
                    d1++;

            edgeDir[0].Average = edgeDir[0].Average / (cuHeight * cuWidth);
            edgeDir[1].Average = edgeDir[1].Average / (cuHeight * cuWidth);
            edgeDir[2].Average = edgeDir[2].Average / (cuHeight * cuWidth);
            edgeDir[3].Average = edgeDir[3].Average / (cuHeight * cuWidth);

            // find max average Sobel value
            for i from 0 to 3
                for j from 0 to 3 - i
                    if edgeDir[j].Average < edgeDir[j + 1].Average
                        tempEdgeDir = edgeDir[j];
                        edgeDir[j] = edgeDir[j + 1];
                        edgeDir[j + 1] = tempEdgeDir;
                    end if

            // in edgeDir[0] max average value from Sobel filters
            if edgeDir[0].dirAngle == 0
                StartMode = 2;
                FinalMode = 18;
            if edgeDir[0].dirAngle == 135
                StartMode = 19;
                FinalMode = 34;
            if edgeDir[0].dirAngle == 90
                StartMode = 35;
                FinalMode = 50;
            if edgeDir[0].dirAngle == 45
                StartMode = 51;
                FinalMode = 66;
        }
}
```

---

Εικόνα 17. Ψευδοκώδικας για τον αλγόριθμο Prop2.

Στο τέλος αυτής της διαδικασίας έχει μειωθεί ο αριθμός των γωνιακών modes που ελέγχει το λογισμικό VTM σε κάθε CU καθ' όλη τη διάρκεια εύρεσης υποψηφίων modes για τη λίστα υποψηφίων καθώς και του βέλτιστου τρόπου χωρισμού των CU στα CTU. Στο επόμενο κεφάλαιο θα αναλυθεί η διαδικασία διεξαγωγής πειραμάτων και θα γίνει η παρουσίαση των αποτελεσμάτων σε πίνακες.

## ΚΕΦΑΛΑΙΟ 6 Πειραματικά Αποτελέσματα

Με σκοπό την εκτίμηση της απόδοσης του αλγορίθμου για τη μείωση της πολυπλοκότητας χρησιμοποιώντας 4 πυρήνες Sobel έγινε ένα σύνολο από δοκιμές σε sequences από τα Common Test Conditions (CTC) σε διαφορετικά Quantization Parameters (QP) για 150 frames με τη βοήθεια ενός υποψήφιου διδάκτορα, τον Παναγιώτη Μπελεμέμη, θέτοντας το λογισμικό αναφοράς σε All Intra (AI) configuration.

Πιο συγκεκριμένα, επιλέχθηκαν 5 sequences από τη Κλάση B των CTC με ανάλυση 1920x1080, δηλαδή Full HD, τα BasketballDrive, Cactus, Kimono, ParkScene και BQTerrace στις Παραμέτρους Κβάντισης (QP) 22, 27, 32 και 37. Για κάθε sequence και QP έγιναν 3 δοκιμές, μία στον πρωτότυπο αλγόριθμο του VTM, μία στον αλγόριθμο Prop1 και μία στον αλγόριθμο Prop2. Επομένως έχουμε  $4 \times 5 \times 3 = 60$  δοκιμές στο σύνολο των πειραμάτων.

Οι δοκιμές έγιναν σε ένα server του Πανεπιστημίου Θεσσαλίας με επεξεργαστή Intel(R) Xeon(R) Silver 4210 CPU χρονισμένο στα 2.20 GHz με 10 cores και 2 threads ανά πυρήνα, με 65610256 kB διαθέσιμη μνήμη RAM και με λειτουργικό Ubuntu 20.04. Το λογισμικό αναφοράς VTM πάνω στο οποίο έγιναν οι αλλαγές στο κώδικα τη στιγμή των δοκιμών ήταν στην εκδοχή 13.0.

Στα αποτελέσματα, με τη βοήθεια του ίδιου του λογισμικού αναφοράς συλλέχθηκε ο ρυθμός αναφοράς (bit rate), το Y-PSNR, το YUV-PSNR και ο συνολικός χρόνος εκτέλεσης. Εφόσον οι αλλαγές που έγιναν αφορούν μόνο το Y στοιχείο του βίντεο, δηλαδή αυτό της φωτεινότητας, επιλέξαμε να συλλέξουμε και το αποτέλεσμα του Y-PSNR. Επιπλέον, για τους δύο προτεινόμενους αλγορίθμους έχει υπολογιστεί η διαφορά τους σε χρόνο εκτέλεσης με τον πρωτότυπο αλγόριθμο του VTM σε ποσοστό επί τοις εκατό (%) για να φανεί η επιτάχυνση ή επιβράδυνση μεταξύ των δύο και έχει ονομαστεί ως Speed-up. Οι παρακάτω πίνακες είναι χωρισμένοι πρώτα στα video sequences στα οποία έγιναν οι δοκιμές και έπειτα στους αλγορίθμους στους οποίους πραγματοποιήθηκαν τα πειράματα.

**Table 1.** Results from the Basket sequence with the Original algorithm.

QP	BasketballDrive, Original algorithm			
	Bit rate (bps)	Y-PSNR (dB)	YUV-PSNR (dB)	Total time (s)
22	63493.9813	41.7802	42.777	96665
27	24019.2027	39.3874	40.477	53421
32	12000.7947	37.827	38.8714	30572
37	6680.0507	36.1272	37.1839	14667

**Table 2.** Results from the Basket sequence with the 1<sup>st</sup> Proposed algorithm.

BasketballDrive, Prop1 algorithm					
<i>QP</i>	<i>Bit rate (bps)</i>	<i>Y-PSNR (dB)</i>	<i>YUV-PSNR (dB)</i>	<i>Total time (s)</i>	<i>Speed-up (%)</i>
22	64212.016	41.7753	42.7726	94071	2.68349454
27	24665.0187	39.3834	40.4729	52848	1.07261189
32	12408.024	37.8081	38.8537	30420	0.49718697
37	6913.032	36.0908	37.1479	15302	-4.32944706

**Table 3.** Results from the Basket sequence with the 2<sup>nd</sup> Proposed algorithm.

BasketballDrive, Prop2 algorithm					
<i>QP</i>	<i>Bit rate (bps)</i>	<i>Y-PSNR (dB)</i>	<i>YUV-PSNR (dB)</i>	<i>Total time (s)</i>	<i>Speed-up (%)</i>
22	63891.2213	41.7761	42.7735	96299	0.378627218
27	24414.024	39.384	40.474	53341	0.149753842
32	12266.0853	37.8163	38.8613	31538	-3.159754023
37	6837.712	36.1054	37.1626	14907	-1.636326447

**Table 4.** Results from the Cactus sequence with the Original algorithm.

Cactus, Original algorithm				
<i>QP</i>	<i>Bit rate (bps)</i>	<i>Y-PSNR (dB)</i>	<i>YUV-PSNR (dB)</i>	<i>Total time (s)</i>
22	100069.9787	41.1296	41.617	115066
27	43101.368	38.2398	38.9422	70078
32	23090.224	36.1092	36.9691	44630
37	12376.8507	33.7546	34.7701	26236

**Table 5.** Results from the Cactus sequence with the 1<sup>st</sup> Proposed algorithm.

Cactus, Prop1 algorithm					
<i>QP</i>	<i>Bit rate (bps)</i>	<i>Y-PSNR (dB)</i>	<i>YUV-PSNR (dB)</i>	<i>Total time (s)</i>	<i>Speed-up (%)</i>
22	101130.3947	41.1259	41.6142	111769	2.86531208
27	44202.9413	38.239	38.9407	67729	3.35197922
32	23840.0507	36.0935	36.9541	44869	-0.53551423
37	12821.9547	33.7193	34.7384	26540	-1.15871322

**Table 6.** Results from the Cactus sequence with the 2<sup>nd</sup> Proposed algorithm.

Cactus, Prop2 algorithm					
<i>QP</i>	<i>Bit rate (bps)</i>	<i>Y-PSNR (dB)</i>	<i>YUV-PSNR (dB)</i>	<i>Total time (s)</i>	<i>Speed-up (%)</i>
22	100736.248	41.1271	41.615	111967	2.693236925
27	43805.0587	38.2392	38.9412	69617	0.657838409
32	23587.3227	36.0995	36.9598	44545	0.190454851
37	12668.4133	33.7337	34.7507	26308	-0.274432078

**Table 7.** Results from the Kimono sequence with the Original algorithm.

Kimono, Original algorithm				
<i>QP</i>	<i>Bit rate (bps)</i>	<i>Y-PSNR (dB)</i>	<i>YUV-PSNR (dB)</i>	<i>Total time (s)</i>
22	17136.4352	43.3994	44.0699	67152
27	9216.5824	42.0542	42.5364	35701
32	5470.7482	40.303	40.8082	21929
37	3251.2371	37.9385	38.5715	10020

**Table 8.** Results from the Kimono sequence with the 1<sup>st</sup> Proposed algorithm.

Kimono, Prop1 algorithm					
<i>QP</i>	<i>Bit rate (bps)</i>	<i>Y-PSNR (dB)</i>	<i>YUV-PSNR (dB)</i>	<i>Total time (s)</i>	<i>Speed-up (%)</i>
22	17264.8934	43.3986	44.0691	65127	3.01554682
27	9303.9066	42.0486	42.5317	34688	2.83745553
32	5528.8986	40.2917	40.799	22182	-1.15372338
37	3289.4042	37.9185	38.5528	10688	-6.66666667

**Table 9.** Results from the Kimono sequence with the 2<sup>nd</sup> Proposed algorithm.

Kimono, Prop2 algorithm					
<i>QP</i>	<i>Bit rate (bps)</i>	<i>Y-PSNR (dB)</i>	<i>YUV-PSNR (dB)</i>	<i>Total time (s)</i>	<i>Speed-up (%)</i>
22	17221.2378	43.3986	44.069	64065	4.597033595
27	9276.1856	42.0511	42.5332	34507	3.344444133
32	5511.145	40.2941	40.8009	21296	2.886588536
37	3280.9894	37.9239	38.5586	10944	-9.221556886

**Table 10.** Results from the ParkScene sequence with the Original algorithm.

ParkScene, Original algorithm				
<i>QP</i>	<i>Bit rate (bps)</i>	<i>Y-PSNR (dB)</i>	<i>YUV-PSNR (dB)</i>	<i>Total time (s)</i>
22	48448.5376	42.1967	42.7516	107145
27	25810.8237	39.1652	39.8642	72852
32	13476.6669	36.1784	37.1275	50077
37	6683.0067	33.3145	34.4148	30015

**Table 11.** Results from the ParkScene sequence with the 1<sup>st</sup> Proposed algorithm.

ParkScene, Prop1 algorithm					
<i>QP</i>	<i>Bit rate (bps)</i>	<i>Y-PSNR (dB)</i>	<i>YUV-PSNR (dB)</i>	<i>Total time (s)</i>	<i>Speed-up (%)</i>
22	48687.9501	42.1953	42.7504	102293	4.52844276
27	25967.9808	39.1582	39.8592	71064	2.45429089
32	13561.4771	36.166	37.1174	48330	3.48862751
37	6731.8579	33.3019	34.4034	29506	1.69581876

**Table 12.** Results from the ParkScene sequence with the 2<sup>nd</sup> Proposed algorithm.

ParkScene, Prop2 algorithm					
<i>QP</i>	<i>Bit rate (bps)</i>	<i>Y-PSNR (dB)</i>	<i>YUV-PSNR (dB)</i>	<i>Total time (s)</i>	<i>Speed-up (%)</i>
22	48616.8742	42.196	42.7515	105045	1.959960801
27	25917.6755	39.16	39.86	72061	1.085762917
32	13540.9421	36.1701	37.121	50435	-0.714899055
37	6721.5014	33.3053	34.4067	30414	-1.329335332

**Table 13.** Results from the Terrace sequence with the Original algorithm.

BQTerrace, Original algorithm				
<i>QP</i>	<i>Bit rate (bps)</i>	<i>Y-PSNR (dB)</i>	<i>YUV-PSNR (dB)</i>	<i>Total time (s)</i>
22	161442.3872	43.1013	43.4485	74992
27	73978.1536	38.1273	39.1434	73373
32	38333.3056	35.3773	36.5736	51153
37	20770.8544	32.8287	34.1217	33842

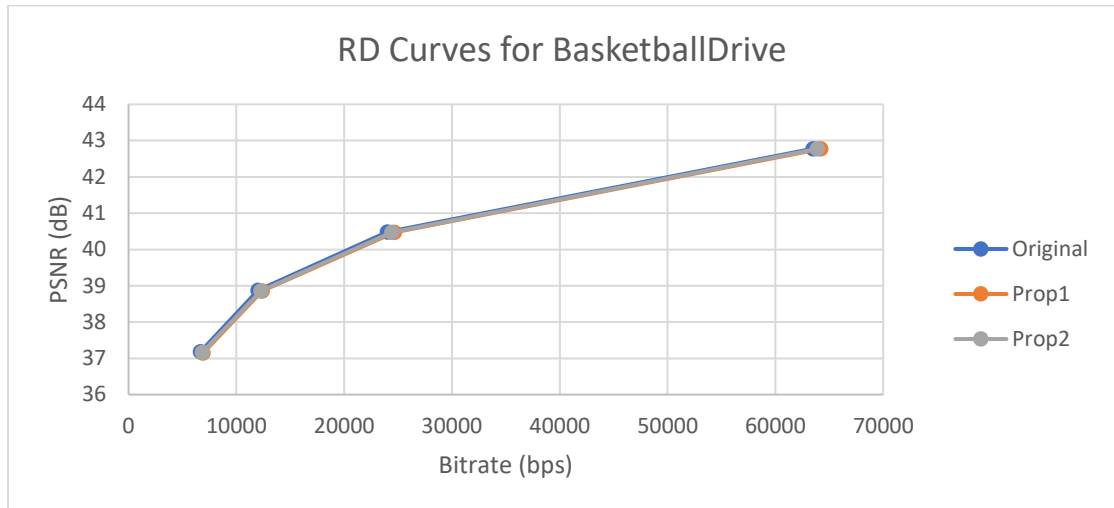
**Table 14.** Results from the Terrace sequence with the 1<sup>st</sup> Proposed algorithm.

BQTerrace, Prop1 algorithm					
<i>QP</i>	<i>Bit rate (bps)</i>	<i>Y-PSNR (dB)</i>	<i>YUV-PSNR (dB)</i>	<i>Total time (s)</i>	<i>Speed-up (%)</i>
22	164687.1008	43.0903	43.4424	75437	-0.59339663
27	76455.4144	38.124	39.1396	72497	1.19389966
32	40087.664	35.3617	36.5583	50815	0.66076281
37	21782.8832	32.7833	34.0795	34048	-0.60871107

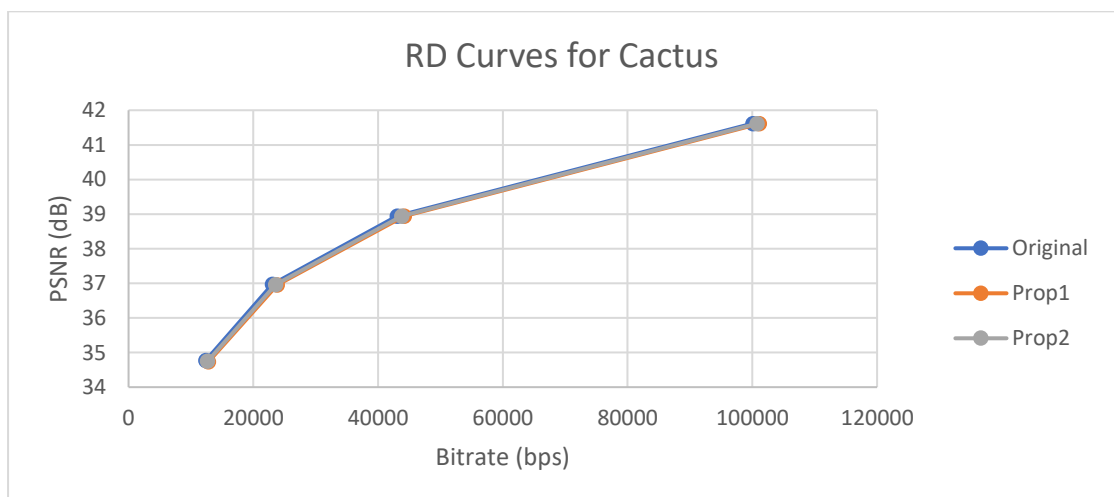
**Table 15.** Results from the Terrace sequence with the 2<sup>nd</sup> Proposed algorithm.

BQTerrace, Prop2 algorithm					
<i>QP</i>	<i>Bit rate (bps)</i>	<i>Y-PSNR (dB)</i>	<i>YUV-PSNR (dB)</i>	<i>Total time (s)</i>	<i>Speed-up (%)</i>
22	162955.1456	43.0946	43.4443	73368	2.165564327
27	75202.0864	38.1232	39.1393	72467	1.234786638
32	39258.0448	35.3662	36.5633	51301	-0.289328094
37	21344.816	32.8021	34.0972	34796	-2.81898233

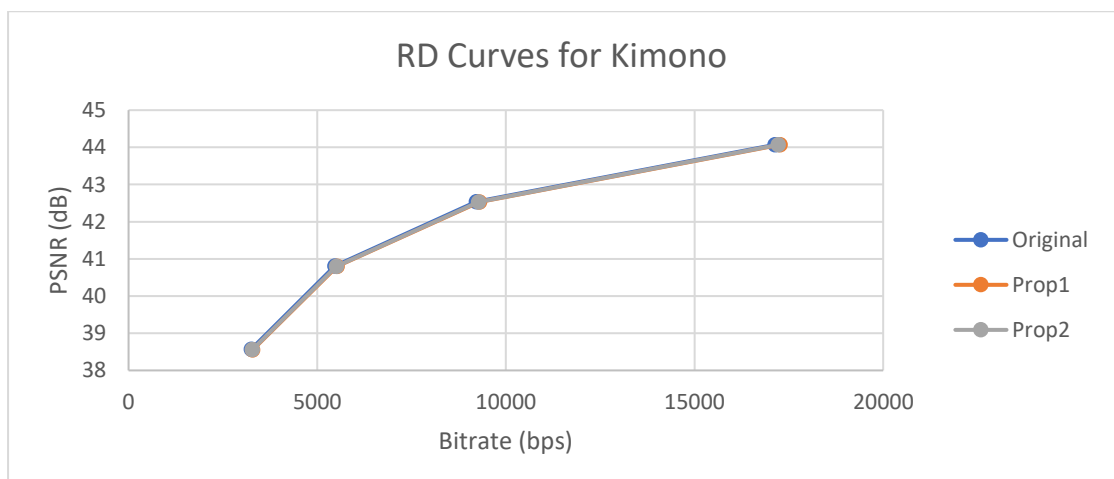
Στη συνέχεια ακολουθούν διαγράμματα με τις καμπύλες RD (RD Curves) των 5 sequence όπου περιέχουν τα αποτελέσματα των τριών αλγορίθμων, του Original, του Prop1 και του Prop2.



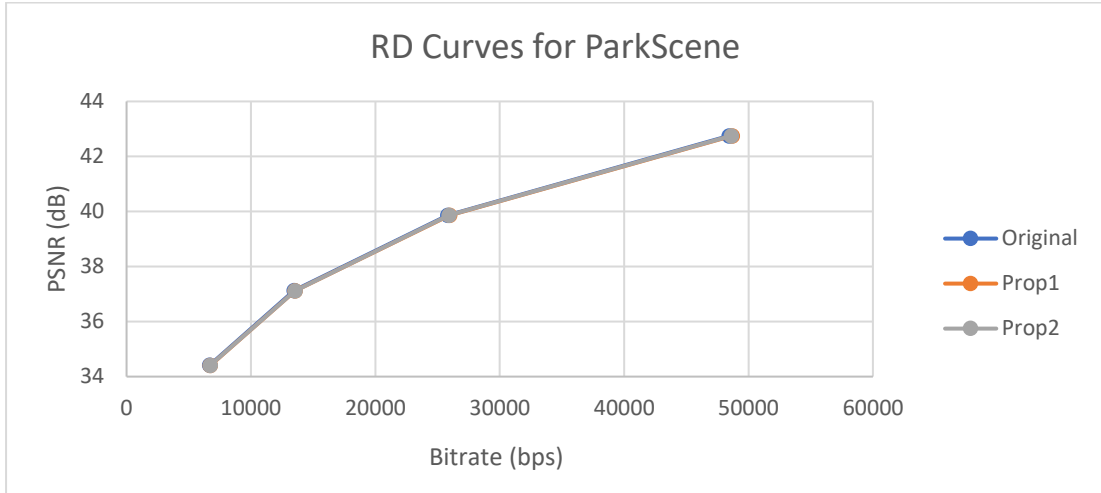
Διάγραμμα 1. Καμπύλες RD των αλγορίθμων για το sequence BasketballDrive.



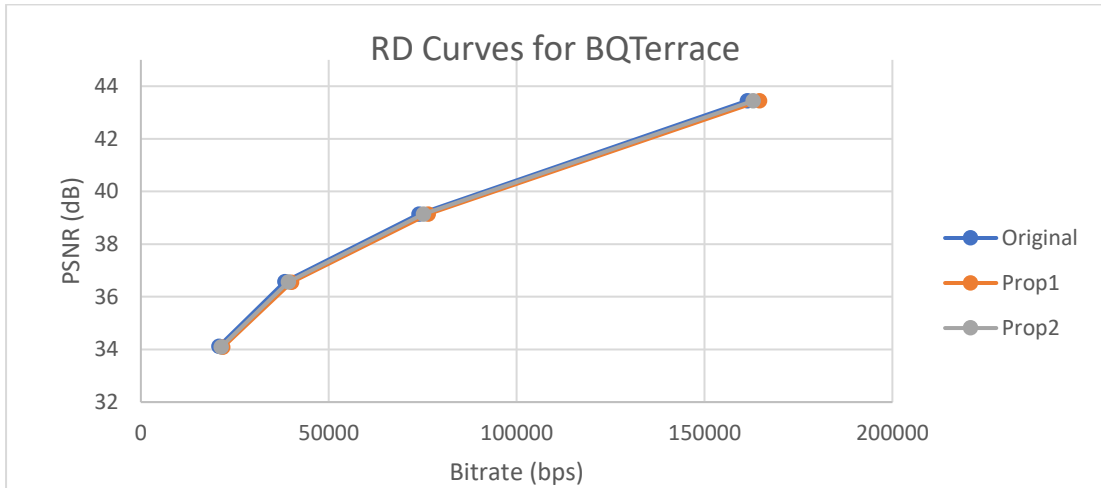
Διάγραμμα 2. Καμπύλες RD των αλγορίθμων για το sequence Cactus.



Διάγραμμα 3. Καμπύλες RD των αλγορίθμων για το sequence Kimono.



Διάγραμμα 4. Καμπύλες RD των αλγορίθμων για το sequence ParkScene.



Διάγραμμα 5. Καμπύλες RD των αλγορίθμων για το sequence BQTerrace.

Εφόσον συλλέχθηκαν τα παραπάνω αποτελέσματα, υπολογίσαμε τις μετρήσεις Bjontegaard Delta (BD) μεταξύ του πρωτότυπου και του Prop1 αλγορίθμου και του πρωτότυπου και του Prop2 αλγορίθμου. Δηλαδή υπολογίσαμε το BD Bit Rate (BDBR) καθώς και το BD Peak Signal-to-Noise Ratio (BDPSNR). Όπως αναφέρθηκε και πριν, η πτυχιακή αυτή αφορά αλλαγές που έγιναν στο Y στοιχείο ενός βίντεο, επομένως υπολογίστηκε και το BDPSNR για τη φωτεινότητα έτσι ώστε να υπάρχει και μία αντικειμενική μέτρηση για το Y στοιχείο.

Table 16. Bjontegaard delta metrics computed between the Original and Prop1 algorithms

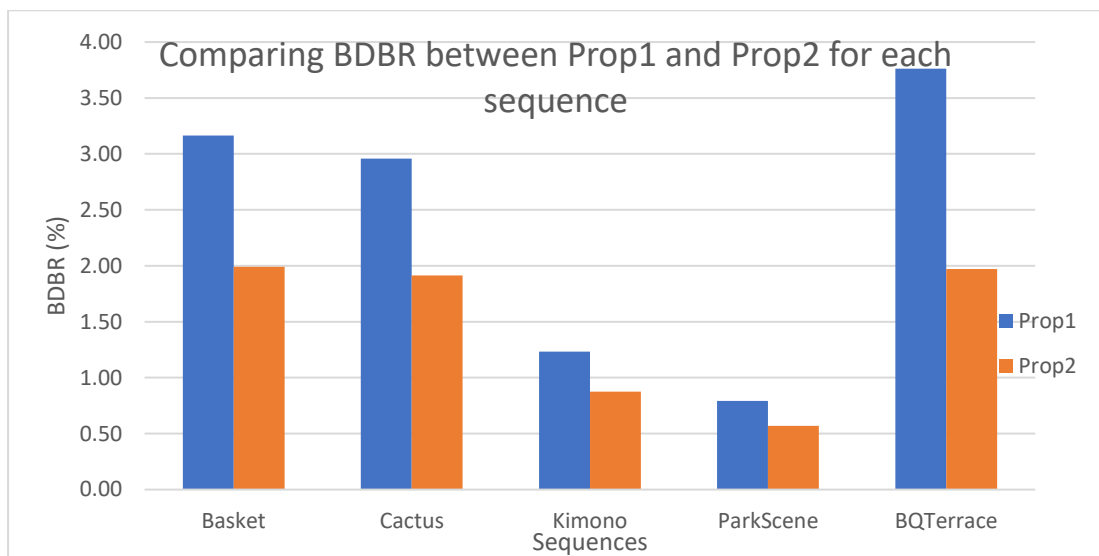
<i>Sequence</i>	BD between the Original and Prop1 algorithms		
	<i>BDBR (%)</i>	<i>Y-BDPSNR (dB)</i>	<i>YUV-BDPSNR (dB)</i>
BasketballDrive	3.16548443	-0.077697859	-0.07736787
Cactus	2.95874368	-0.102825639	-0.09574386
Kimono	1.23247081	-0.042026056	-0.04074153
ParkScene	0.79250392	-0.03674687	-0.03324671
BQTerrace	3.76117689	-0.186059145	-0.17136439



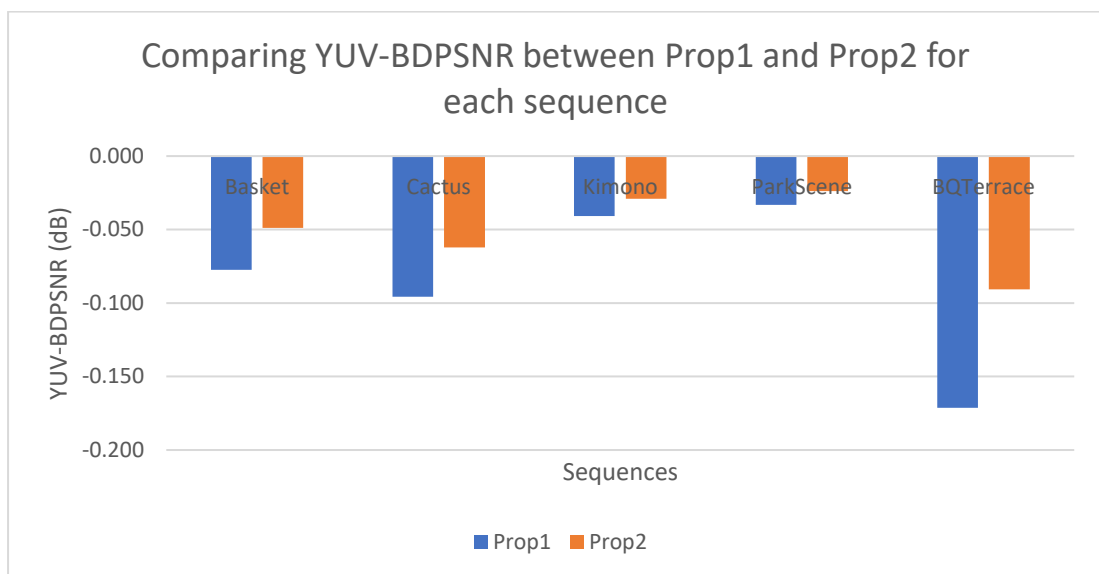
**Table 17.** Bjontegaard delta metrics computed between the Original and Prop2 algorithms

<i>Sequence</i>	BD between the Original and Prop2 algorithms		
	<i>BDBR (%)</i>	<i>Y-BDPSNR (dB)</i>	<i>YUV-BDPSNR (dB)</i>
BasketballDrive	1.992551865	-0.049436304	-0.048945815
Cactus	1.91459347	-0.066620765	-0.062123396
Kimono	0.876750557	-0.029782005	-0.029006974
ParkScene	0.570513611	-0.026462429	-0.023944169
BQTerrace	1.971809931	-0.098419569	-0.090666878

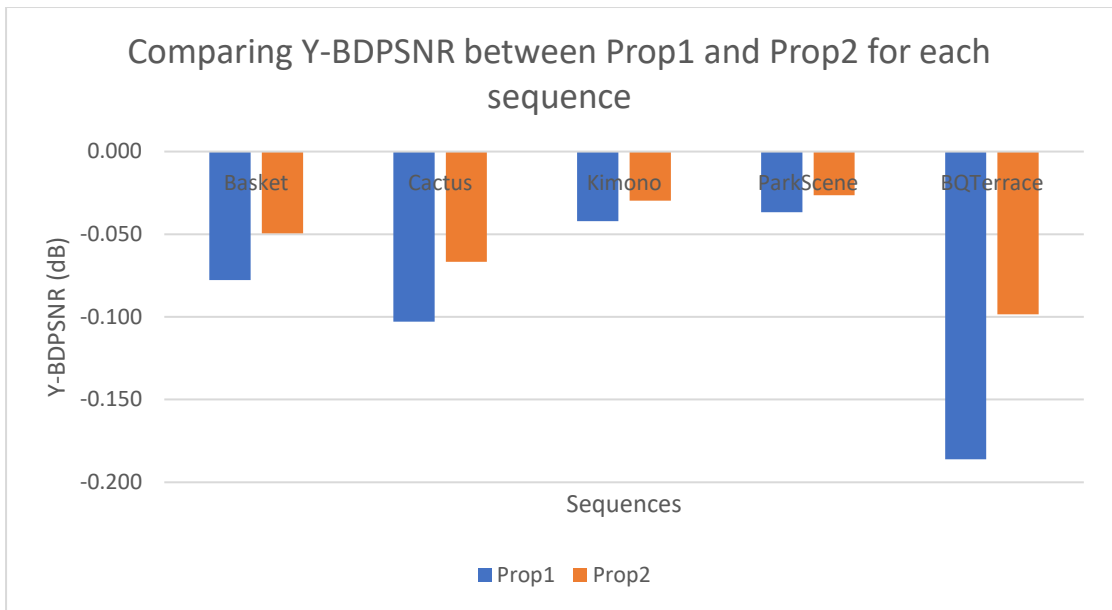
Στη συνέχεια ακολουθούν γραφήματα που συγκρίνουν την απόδοση μεταξύ των δύο αλγορίθμων όσον αφορά τις μετρήσεις BD Bit Rate, Y-BD PSNR και YUV-BD PSNR.



**Διάγραμμα 6.** Γράφημα σύγκρισης BDBR μεταξύ των αλγορίθμων Prop1 και Prop2.



**Διάγραμμα 7.** Γράφημα σύγκρισης YUV-BDPSNR μεταξύ των αλγορίθμων Prop1 και Prop2.



**Διάγραμμα 8.** Γράφημα σύγκρισης Y-BDPSNR μεταξύ των αλγορίθμων Prop1 και Prop2.

## ΚΕΦΑΛΑΙΟ 7 Συμπεράσματα

---

Σε αυτή τη πτυχιακή παρουσιάστηκε το νέο πρότυπο συμπίεσης βίντεο VVC, μαζί με τα νέα χαρακτηριστικά που προσφέρει σε σχέση με τα προηγούμενα πρότυπα. Στη συνέχεια, εφόσον έγινε μία αναφορά προηγούμενων ερευνητικών εργασιών στη βιβλιογραφική επισκόπηση, παρουσιάστηκε ο τελεστής Sobel που εφαρμόστηκε πάνω στο λογισμικό αναφοράς VTM σε έναν αλγόριθμο με σκοπό τη μείωση της πολυπλοκότητας του κωδικοποιητή του VVC στη πρόβλεψη Intra-frame. Έπειτα από την υλοποίηση του στο VTM, πραγματοποιήθηκαν ένα σύνολο πειραμάτων σε προτεινόμενα sequences από τα CTC με στόχο την εκτίμηση της απόδοσής του υπολογίζοντας τις μετρήσεις Bjontegaard Delta.

Μετά την εφαρμογή του αλγορίθμου στο λογισμικό αναφοράς και τα πειράματα που διεξήχθησαν, από τα αποτελέσματα μπορούμε να συμπεράνουμε ότι από τους αλγορίθμους Prop1 και Prop2 ο δεύτερος προσφέρει καλύτερα αποτελέσματα. Πιο συγκεκριμένα, αποδίδει καλύτερα στα χαμηλά QPs στα οποία πραγματοποιήθηκαν τα πειράματα (22 και 27 QP) φέροντας μία μικρή μείωση στη ποιότητα καθώς και μία μικρή αύξηση στο bitrate, μέσω των μετρήσεων BDPSNR και BDBR αντίστοιχα, ενώ ταυτόχρονα προσφέρει καλύτερη επιτάχυνση. Αυτό είναι λογικό καθώς όσο μικρότερο είναι το QP τόσο μικρότερη είναι και η απόκλιση των δεδομένων στα οποία εφαρμόζονται τα φίλτρα Sobel από τα αρχικά δεδομένα της εικόνας. Παρ' όλα αυτά, η επιτάχυνση που προσφέρεται είναι σχετικά μικρή, επομένως ο περιορισμός των γωνιακών mode παρέχει περιορισμένη δυνατότητα μείωσης της συνολικής πολυπλοκότητας του κωδικοποιητή.

Σύμφωνα με τα παραπάνω, υπάρχουν ευκαιρίες επιτάχυνσης του λογισμικού όπως και μείωσης της πολυπλοκότητάς του. Αρχικά, μία προέκταση του αλγορίθμου είναι η εφαρμογή και υπολογισμός των φίλτρων Sobel βάσει της τιμής κβαντισμού. Επιπλέον, όπως προτάθηκε και από μερικές ερευνητικές εργασίες στη βιβλιογραφική ανασκόπηση, μία προέκταση θα ήταν η χρήση των φίλτρων Sobel σε άλλες διαδικασίες της κωδικοποίησης όπως αυτής του διαμερισμού των CU υπολογίζοντας την υφή και έπειτα διακόπτοντας νωρίτερα τον διαχωρισμό, μειώνοντας σημαντικά κατ' αυτόν το τρόπο τη πολυπλοκότητα. Τέλος, μπορεί να προστεθεί και η εφαρμογή άλλων τεχνικών που έχουν εφαρμογή στην ψηφιακή επεξεργασία εικόνας, όπως το κριτήριο της διακύμανσης.

## ΕΕΝΗ ΒΙΒΛΙΟΓΡΑΦΙΑ

---

- [1] Video Coding Standardization GitLab 2021, VVCSoftware\_VTM, Fraunhofer Heinrich Hertz Institute, [https://vcgit.hhi.fraunhofer.de/jvet/VVCSoftware\\_VTM](https://vcgit.hhi.fraunhofer.de/jvet/VVCSoftware_VTM)
- [2] ISO/IEC JTC 1/SC 29 2020, H.266, International Telecommunications Union, <https://www.itu.int/rec/T-REC-H.266>
- [3] ISO/IEC JTC 1/SC 29 2021, ISO/IEC 23090-3:2021, International Organization for Standardization, <https://www.iso.org/standard/73022.html>
- [4] newsroom, D., 2021-2025, I., & President-elect, A. (2021). What we do. Retrieved 22 June 2021, from <https://www.iso.org/what-we-do.html>
- [5] Standards. (2021). Retrieved 22 June 2021, from <https://www.iso.org/standards.html>
- [6] Members. (2021). Retrieved 22 June 2021, from <https://www.iso.org/members.html>
- [7] ISO/IEC JTC 1 2014, ISO/IEC JTC 1: Vision, Mission and Principles, [https://www.iso.org/files/live/sites/isoorg/files/developing\\_standards/docs/en/jtc1\\_mission\\_brochure\\_2014\\_final.pdf](https://www.iso.org/files/live/sites/isoorg/files/developing_standards/docs/en/jtc1_mission_brochure_2014_final.pdf)
- [8] ISO/IEC JTC 1/SC 29 1991, Coding of audio, picture, multimedia and hypermedia information, <https://www.iso.org/committee/45316.html>
- [9] JVET - Joint Video Experts Team. (2021). Retrieved 1 July 2021, from <https://www.itu.int/en/ITU-T/studygroups/2017-2020/16/Pages/video/jvet.aspx>
- [10] About ITU. (2021). Retrieved 1 July 2021, from <https://www.itu.int/en/about/Pages/default.aspx>
- [11] Frequently Asked Questions | IEC. (2021). Retrieved 1 July 2021, from <https://www.iec.ch/faq>
- [12] Video Coding Experts Group (VCEG). (2021). Retrieved 1 July 2021, from <https://www.itu.int/en/ITU-T/studygroups/2017-2020/16/Pages/video/vceg.aspx>
- [13] N17195, Joint Call for Proposals on Video Compression with Capability beyond HEVC | MPEG. (2021). Retrieved 31 July 2021, from

<https://mpeg.chiariglione.org/standards/exploration/future-video-coding/n17195-joint-call-proposals-video-compression-capability>

- [14] Bitmovin. (2019). *Video Developer Report 2019* [Ebook]. Retrieved from <https://cdn2.hubspot.net/hubfs/3411032/Bitmovin%20Magazine/Video%20Developer%20Report%202019/bitmovin-video-developer-report-2019.pdf>
- [15] Dong, X., Shen, L., Yu, M., & Yang, H. (2021). Fast Intra Mode Decision Algorithm for Versatile Video Coding. *IEEE Transactions On Multimedia*, 1-1. doi: 10.1109/tmm.2021.3052348
- [16] BenHajyoussef, A., Ezzedine, T., & Bouallègue, A. (2017). Gradient-based pre-processing for intra prediction in High Efficiency Video Coding. *EURASIP Journal On Image And Video Processing*, 2017(1). doi: 10.1186/s13640-016-0159-9
- [17] Belghith, F., Kibeya, H., Ben Ayed, M., & Masmoudi, N. (2015). Fast coding unit partitioning method based on edge detection for HEVC intra-coding. *Signal, Image And Video Processing*, 10(5), 811-818. doi: 10.1007/s11760-015-0820-2
- [18] Yang, H., Shen, L., Dong, X., Ding, Q., An, P., & Jiang, G. (2020). Low-Complexity CTU Partition Structure Decision and Fast Intra Mode Decision for Versatile Video Coding. *IEEE Transactions On Circuits And Systems For Video Technology*, 30(6), 1668-1682. doi: 10.1109/tcsvt.2019.2904198
- [19] J. Chen, H. Sun, J. Katto, X. Zeng and Y. Fan, "Fast QTMT Partition Decision Algorithm in VVC Intra Coding based on Variance and Gradient," 2019 IEEE Visual Communications and Image Processing (VCIP), 2019, pp. 1-4, doi: 10.1109/VCIP47243.2019.8965674.