



## Utilizing Median Filter as Pre-processing for Depth Map Compression

Christin Erniati Panjaitan

Institut Teknologi Del (IT-DEL), email: [christin.panjaitan@del.ac.id](mailto:christin.panjaitan@del.ac.id)

### Abstrak

*Kompresi video telah banyak dikembangkan dan sudah banyak standarisasi yang dibuat oleh industri-industri multimedia. Sebuah format 3D terdiri dari tradisional 2D video dan depth. Untuk membentuk 3D video, kompresi video depth dibutuhkan untuk memberikan informasi geometris yang kelaknya akan digunakan untuk sintesis pandangan maya. Kompresi video yang tradisional tidak bisa dipergunakan langsung ke video depth yang asli karena karakteristiknya yang akan memberikan efek pada kompresi ratio dan tampilan sintesis itu sendiri. Informasi depth dapat dipergunakan untuk beberapa aplikasi pengolahan gambar seperti identifikasi, segmentasi dan lain-lain. Adapun kerusakan yang terjadi pada video depth asli umumnya berasal dari proses awal akuisisi video depth. Video depth ini perlu dikompres untuk efisiensi jaringan transmisi. Di dalam penelitian kali ini akan didemonstrasikan penggunaan median filter sebagai metode pre-processing. Keluaran dari pre-preprocessing akan dikirim ke kompresi H.264/AVC. Dari data-data penelitian yang diperoleh, penggunaan median filter sebagai teknik pre-processing telah meningkatkan kualitas kompresi video 16-20 % dibandingkan dengan yang tidak menggunakan pre-processing sama sekali.*

**Kata kunci:** Kompresi Video, Pre-processing, H.264/AVC

### Abstract

*Video codings are widely explored, and many standards have been established from multimedia industries. The 3D video format consists of the conventional 2D video format and depth maps. In order to support 3D video, depth video compression is needed to reveal geometric information for synthesis virtual views. Conventional video coding is not recommended directly employed into original depth video because its characteristics which can influence in compression ratio and synthesis performance. Depth information is able to be used for some image processing applications such as identification, segmentation and so on. Generally, the damage occurs in original depth video begins from acquisition process depth video. This depth video is needed to be compressed for efficient network transmission. In this research, will be shown utilizing median filter as pre-processing method. Then, the output of pre-processing is fed into H.264/AVC encoding tool. From experimental results, utilizing median filter as a pre-processing technique already enrich the quality of video compression 16 % to 20 % compared without using pre-processing at all.*

**Keywords:** Video Coding, Pre-processing, Median Filter, H.264/AVC

## 1. Pendahuluan

Video 3D (*3DV*) merupakan kumpulan dari sinyal video 2D dan depth. Diiringin dengan perkembangan monitor 3D dan sistem interaktif multimedia, maka banyak kementerian dari dunia industri dan akademisi untuk mengembangkan 3D ke beberapa aplikasi seperti TV 3D (3D Television) dan sentuhan point bebas (*Free Viewpoint Video*).

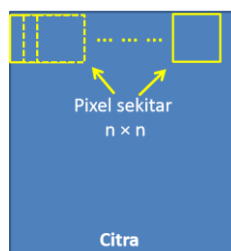
Video depth terdiri dari 8 bit atau 16 bit yang dianggap sebagai tipe ukuran abu-abu (grayscale). Video depth dapat dihasilkan dari kamera yang telah memiliki sensor depth seperti : Microsoft Kinect, Time-of-Flight dan Scanner laser. Dengan adanya informasi depth ini maka akan banyak sekali aplikasi yang bisa dikerjakan seperti identifikasi obyek, segmentasi objek dari background (foreground/background segmentation), relighting gambar (image relighting) dan lain-lain. Sayangnya, video depth sering bermasalah karena rendahnya resolusi, kebisingan (noise), timbulnya bagian asing (outliers) dan beberapa lubang (holes) [1-3].

Atas pemikiran di atas, maka penelitian kali ini ingin menggunakan median filter sebagai teknik pre-processing. Median filter mampu memberikan pengurangan kebisingan (noise reduction) dan cukup terkenal untuk mengatasi kebisingan acak (random noise). Saat ini banyak modifikasi filter yang digunakan dari bilateral filter untuk membersihkan beberapa noise sebelum masuk ke tahapan pengolahan utama. Hanya saja, bilateral filter membutuhkan waktu yang cukup lama karena dua filter bekerja secara paralel. Pada penelitian kali ini, video depth akan difilter menggunakan median filter dan keluarannya akan dikirim ke kompresi H.264/AVC [4, 6].

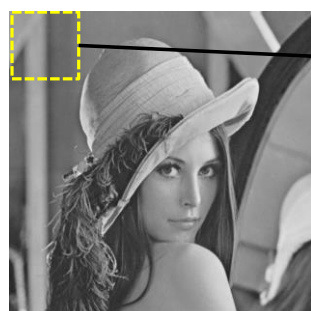
## 2. Dasar Teori

### 2.1 Median Filter

Filter order statik adalah non-linear filter ruang yang bertanggungjawab mengurutkan piksel-piksel pada area gambar yang mencakup dari filter, dan mengganti nilai tengah piksel dengan nilai yang ditentukan dari hasil order. Contohnya jikalau ada  $3 \times 3$  tetangga (neighborhood) maka mediannya adalah urutan ke 5. Jikalau ada  $5 \times 5$  tetangga maka mediannya adalah urutan ke 13 seperti yang ditunjukkan pada gambar 1 dan gambar 2. Jadi, prinsip kerja median filter adalah memaksa point tertentu untuk menjadi linear dengan nilai-nilai tetangganya [4].



Gambar 1. Konsep Median Filter



$$X_{3 \times 3} = [25, 26, 26, 24, 23, 27, 24, 28, 23]$$



$$X_{3 \times 3}' = [23, 23, 24, 24, 25, 26, 26, 27, 28]$$

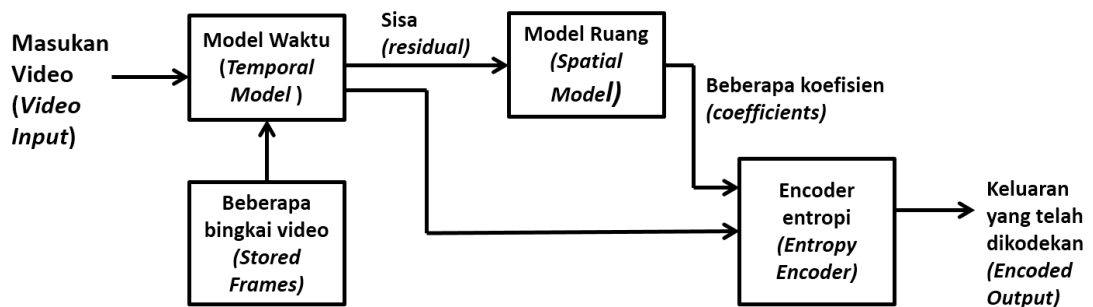


$$X_{3 \times 3} = [25, 26, 26, 24, 25, 27, 24, 28, 23]$$

Gambar 2. Contoh Implementasi Median Filter Dengan Neighborhood 3x3

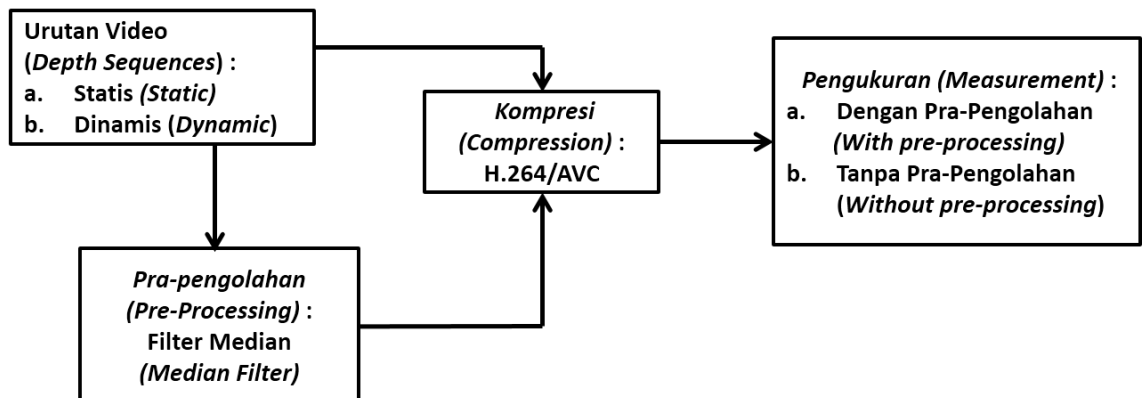
## 2.2 Kompresi Video H.264/AVC

Seperti yang didemonstrasikan di gambar 3 bahwa kompresi video terdiri atas 3 bagian yakni : model waktu (temporal), model ruang (spatial), dan model entropi. Model waktu mencoba mengurangi redundansi waktu dengan mengeksplor kemiripan frame demi frame. Pada H.264, prediksi ini dibentuk dari satu atau banyak frame sebelum atau sesudahnya dan lagi ditingkatkan dengan kompensasi. Keluaran dari model waktu adalah sisa frame yang akan merupakan masukan untuk model ruang. Pada model ruang akan dieksplor kemiripan antar ruang di dalam frame untuk mengurangi redundansi pada frame itu sendiri. Pada H.264, hal ini dapat dicapai dengan mengimplementasikan transformasi ke sample-sample residual dan mengkuantisasi hasilnya. Di transformasi, akan dikonvert sample-sample tersebut ke suatu domain dan akan direpresentasikan dengan koefisien transformasi. Koefisien-koefisien yang ada akan dikuantisasi untuk menghilangkan nilai-nilai yang tidak signifikan. Sehingga keluaran dari model ruang adalah koefisien transformasi yang telah dikuantisasi yang nantinya akan dikompres dengan encoder entropi [5].



Gambar 3. Blok diagram encoder video

## 3. Sistem Kerja



Gambar 4. Blok diagram penelitian

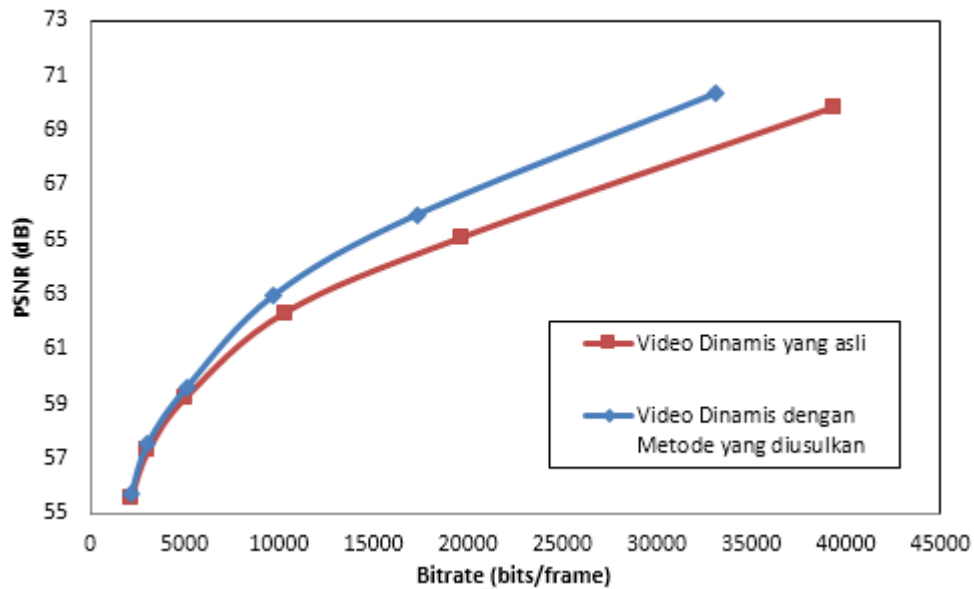
Video depth yang diperoleh dari Microsoft Kinect langsung dikompres dengan H.264/AVC dan akan dibandingkan kualitasnya dengan video depth yang menggunakan pre-processing sebelum dikirim ke H.264/AVC. Adapun kuantisasi parameter [QP] yang digunakan adalah 5, 10, 15, 20, 25 dan 30. Video yang dianalisa adalah video statis, yang objeknya tidak memiliki pergerakan dari awal hingga akhir frame dan video dinamis yang objeknya ada pergerakan dari awal hingga akhir frame.

#### 4. Hasil Penelitian

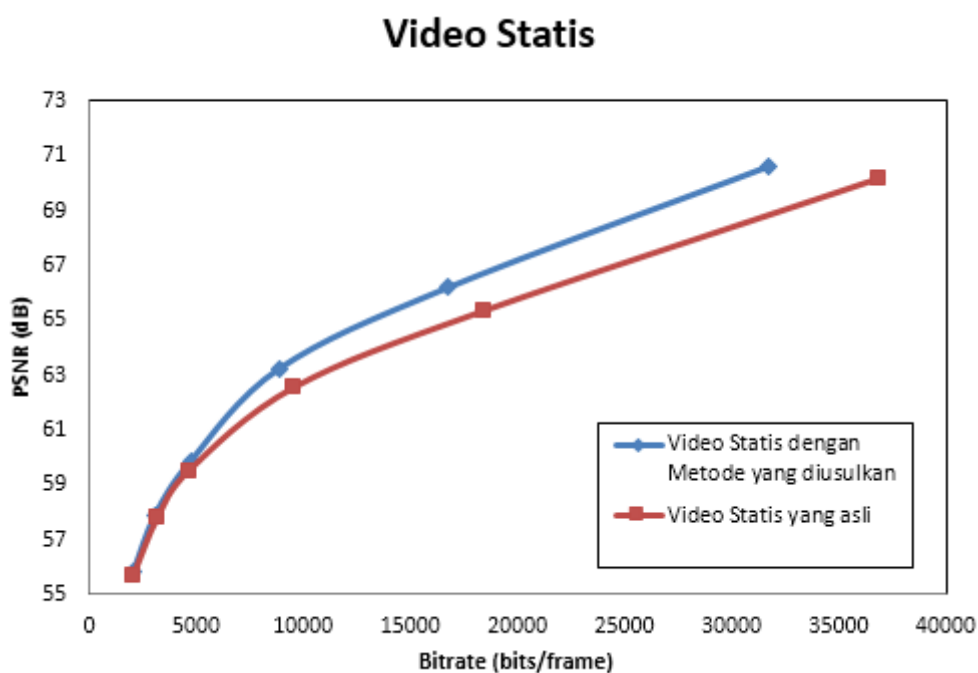
Tabel 1. Perbandingan Rate-Distortion

Uraian	Video asli Dinamis Depth (Original Depth Dynamic Video)		Video Dinamis dengan Metode yang diusulkan (Dynamic Proposed Method)		Video asli Statis Depth (Original Depth Static Video)		Video Statis dengan Metode yang diusulkan (Static Proposed Method)	
	PSNR (dB)	Bitrate (bits/frame)	PSNR (dB)	Bitrate (bits/frame)	PSNR (dB)	Bitrate (bits/frame)	PSNR (dB)	Bitrate (bits/frame)
QP = 5	69.816	39333	70.345	33082	70.163	36824	70.627	31703
QP = 10	65.054	19604	65.904	17263	65.358	18459	66.213	16782
QP = 15	62.323	10282	62.961	9684	62.565	9576	63.256	8887
QP = 20	59.249	5065	59.598	5075	59.513	4645	59.883	4746
QP = 25	57.296	3042	57.516	2985	57.807	3191	57.884	3057
QP = 30	55.546	2110	55.688	2110	55.709	2071	55.825	2047

#### Video Dinamis



Gambar 5 Rate Distortion Video Dinamis



**Gambar 6 Rate Distortion Video Statis**

Pada Tabel 1 diperlihatkan data perbandingan hasil kompresi antara video yang dinamis dan statis. Gambar 5 dan 6 adalah visualisasi dari data pada tabel 1. Garis merah menginformasikan video depth yang tidak menggunakan pre-processing dan garis biru menginformasikan video depth yang menggunakan pre-processing terlebih dahulu sebelum dikirim ke H.264/AVC.

Pada Gambar 5, dapat dilihat representasi bits/frame pada video dinamis yang original sekitar 4000 bits/frame dengan PSNR 69 dB sedangkan untuk video yang menggunakan pre-processing memiliki bits/frame sekitar 3200 bits/frame dengan PSNR sekitar 70 dB. Pada gambar 6, dapat dilihat representasi bits/frame pada original video statis sekitar 3700 bits/frame dengan PSNR sekitar 69 dB dan yang menggunakan pre-processing pada bits/frame sekitar 3100 memiliki PSNR sekitar 70 dB. Dari representasi data tersebut, dapat diidentifikasi bahwa video dinamis memiliki informasi yang lebih banyak dibandingkan video statis. Video yang memiliki pre-processing akan memiliki kualitas kompresi yang baik dibandingkan yang tidak memiliki pre-processing.

## 5. Kesimpulan

Video depth memiliki kualitas yang buruk disebabkan noise yang diperoleh saat proses akuisisi. Untuk meningkatkan kualitas kompresi yang baik diperlukan tahapan pre-processing yang bertujuan untuk memperbaiki video depth.

## Daftar Pustaka

- [1] Shujie Liu, PoLin Lai, Dong Tian, Chang Wen Chen, "New Depth Coding Techniques With Utilization of Corresponding Video", IEEE Transactions on Broadcasting, Vol.57, No.2, June 2011
- [2] Lu Shen, King Ngi Ngan, C-L. Lim, Songnan Li, "Online Temporally Consistent Indoor Depth Video Enhancement via Static Structure", IEEE Transactions on Image Processing, Vol.24, No.7, July 2015

- [3] Ismael Daribo, Dinei Florencio, Gene Cheung, "Arbitrarily Shaped Motion Prediction for Depth Video Compression Using Arithmetic Edge Coding", IEEE Transactions on Image Processing, Vol.23, No.11, November 2014
- [4] R. C. Gonzalez, R. E. Woods, "Digital Image Processing", Pearson Prentice Hall, 3rd Edition, 2008.
- [5] Iain E. G. Richardson, "H.264 and MPEG-4 Video Compression", John Willey & Sons Ltd, 2003
- [6] Tomasi C, Manduchi R (1998) Bilateral filtering for gray and color images. In proceedings 6<sup>th</sup> International Conference on Computer Vision, pp 839-846