

**UNIVERSITATEA DIN BUCUREŞTI**  
**FACULTATEA DE MATEMATICA SI INFORMATICA**  
**SPECIALIZAREA INFORMATICA**

**Proiect Procesarea Semnalelor**

**Reducerea zgomotului in video-uri prin filtrare  
spatio-temporală**

**Student:**

Agușoaiei Alexandru-Gabriel

## **Rezumat**

Această lucrare analizează metode moderne și clasice de reducere a zgomotului în secvențe video, evidențiind avantajele filtrării spațio-temporale față de abordările 2D. Sunt prezentate modele matematice ale semnalului video, tipuri de zgomot și artefacte temporale, precum și o clasificare detaliată a algoritmilor de filtrare, de la filtre liniare simple până la metode bazate pe rețele neurale.

## **Abstract**

This paper analyzes classical and modern video denoising methods, highlighting the advantages of spatio-temporal filtering over frame-by-frame approaches. Mathematical models of video signals, noise types, temporal artifacts, and a detailed classification of filtering algorithms are presented, ranging from linear filters to deep learning-based approaches.

# Cuprins

<b>1</b>	<b>Introducere</b>	<b>1</b>
1.1	Filtrare 2D vs Filtrare Spațio-Temporală . . . . .	1
<b>2</b>	<b>Fundamente teoretice</b>	<b>1</b>
2.1	Modelarea Semnalului Video . . . . .	1
2.2	Coerenta Temporală . . . . .	1
2.3	Artefacte specifice . . . . .	1
<b>3</b>	<b>Clasificare Generală</b>	<b>1</b>
3.1	Filtre Spațio-Temporale Liniare . . . . .	2
3.1.1	Medierea Temporală . . . . .	2
3.1.2	Filtrul Gaussian 3D . . . . .	2
3.2	Filtre Spatio-Temporale Adaptive . . . . .	3
3.2.1	Filtrul Median 3D . . . . .	3
3.2.2	Filtrarea KNN Temporală (k-Nearest Neighbors) . . . . .	4
3.2.3	Filtre Adaptive la Miscare (Motion Adaptive Filtering) . . . . .	4
3.3	Metode Non-Local (Non-Local Means) . . . . .	5
3.3.1	Principiul Auto-similarității . . . . .	5
3.4	Filtrare cu Compensarea Mișcării . . . . .	5
3.4.1	V-BM3D (Video Block-Matching and 3D Filtering) . . . . .	5
3.5	Metode bazate pe Deep Learning . . . . .	6
3.5.1	Abordari Multi-Frame fară Aliniere Explicita . . . . .	6
3.5.2	FastDVDNet . . . . .	7
<b>4</b>	<b>Experimente și Evaluare</b>	<b>7</b>
4.1	Metrici de Evaluare . . . . .	7
4.1.1	PSNR (Peak Signal-to-Noise Ratio) . . . . .	7
4.1.2	SSIM (Structural Similarity Index) . . . . .	8
4.2	Setul de Date și Metodologie . . . . .	8
4.3	Rezultate Experimentale . . . . .	8
4.3.1	Scenariul Static . . . . .	8
4.3.2	Scenariu Dinamic (Fotbal) . . . . .	9
4.3.3	Analiza stabilității Temporale . . . . .	9
<b>5</b>	<b>Concluzii</b>	<b>10</b>

# 1 Introducere

Zgomotul reprezintă una din cele mai frecvente degradări ale semnalului video, apărând în special în condiții slabe de iluminare sau din cauza limitelor hardware ale senzorilor. Spre deosebire de imaginile statice, semnalul video conține redundanță temporală, care poate fiexploata pentru a obține rezultate superioare de restaurare.

## 1.1 Filtrare 2D vs Filtrare Spațio-Temporală

Filtrarea 2D aplicată independent fiecărui cadru ignoră corelațiile temporale și conduce frecvent la apariția artefactului de *flickering*, percepție ca o variație neplăcută a intensităților în timp. Filtrarea spațio-temporală tratează voideoclip-ul ca un volum tridimensional, asigurând coerentă temporală.

# 2 Fundamente teoretice

## 2.1 Modelarea Semnalului Video

Un semnal video discret poate fi descris ca:

$$V(x, y, t) = S(x, y, t) + N(x, y, t)$$

unde  $S$  este semnalul ideal, iar  $N$  reprezintă zgomotul adăugat.

## 2.2 Coerență Temporală

În scene naturale, modificările dintre cadre succesive sunt limitate, iar această proprietate este exploata pentru separarea zgomotului de structurile reale.

## 2.3 Artefacte specifice

- **Flickering** - lipsa consistenței temporale
- **Ghosting** - mediere temporală fără compensarea mișcării

# 3 Clasificare Generală

Metodele de filtrare video pot fi clasificate astfel:

1. Filtre spațio-temporale liniare
2. Filtre spațio-temporale adaptive
3. Metode Non-Local bazate pe patch-uri
4. Metode cu compensarea mișcării
5. Metode bazate pe Deep Learning

### 3.1 Filtre Spațio-Temporale Liniare

#### 3.1.1 Medierea Temporală

Această metodă se bazează pe presupunerea ca zgomotul  $N(x, y, t)$  este o variabilă aleatoare cu medie zero și este independentă și identic distribuită (i.i.d) în timp. Estimarea semnalului filtrat  $\hat{V}$  se obține prin media aritmetică a  $K$  cadre (fereastră temporală):

$$\hat{V}(x, y, t) = \frac{1}{K} \sum_{i=-k}^k V(x, y, t + i)$$

Din punct de vedere statistic, dacă zgomotul are deviația standard  $\sigma_n$ , procesul de mediere reduce deviația standard a zgomotului rezidual la:

$$\sigma_{residual} = \frac{\sigma_n}{\sqrt{K}}$$

Aceasta demonstrează că raportul semnal-zgomot (SNR) crește proporțional cu radăcina pătrată a numărului de cadre utilizate

**Limitări:** Deși eficientă computațional, metoda este ideală doar pentru scene statice. Orice mișcare a obiectelor în scenă incalcă ipoteza de coerență temporală, generând artefacte de tip *Ghosting* (urme semitransparente), deoarece pixelii din cadre diferite nu mai corespund aceleiași structuri fizice.

#### 3.1.2 Filtrul Gaussian 3D

Spre deosebire de mediere simplă, filtrul Gaussian 3D aplică o convoluție cu un nucleu tri-dimensional, acordând o importanță mai mare vecinilor spațio-temporali imediați. Aceasta este

separabil, putând fi descompus în filtre 1D (temporal) și 2D (spatial):

$$G(x, y, t) = \underbrace{\frac{1}{2\pi\sigma_s^2} e^{-\frac{x^2+y^2}{2\sigma_s^2}}}_{\text{Spatial}} \cdot \underbrace{\frac{1}{\sqrt{2\pi}\sigma_t^2} e^{-\frac{t^2}{2\sigma_t^2}}}_{\text{Temporal}}$$

Unde:

- $\sigma_s$  controlează gradul de netezire spațială (blurring).
- $\sigma_t$  controlează extinderea influenței temporale.

Avantajul major față de medierea simplă este atenuarea efectului de *ringing* în domeniul frecvență și o tranziție mai lină a obiectelor în mișcare (ghosting-ul este mai difuz, deci mai puțin deranjant vizual), deși nu este eliminat complet.

## 3.2 Filtre Spatio-Temporale Adaptive

Principala limitare a filtrelor liniare este caracterul lor izotrop: ele aplică aceeași operație de netezire indiferent dacă zona procesată este o regiune omogenă, o muchie sau o zonă cu mișcare rapidă. Acest lucru duce la pierderea detaliilor fine și la artefacte de *blurring*.

Filtrele adaptive încearcă să rezolve această problemă prin ajustarea ponderilor sau a ferestrelor de filtrare în funcție de caracteristicile locale ale semnalului video.

### 3.2.1 Filtrul Median 3D

Filtrul Median 3D este un filtru neliniar, bazat pe statistici de ordin (*order statistics*). Pentru fiecare voxel  $(x, y, t)$ , se consideră o vecinătate spatio-temporală  $\Omega$ , iar noua valoare este determinată prin sortarea tuturor din  $\Omega$  și alegerea elementului central:

$$\hat{V}(x, y, t) = \text{median}\{V(i, j, k) \mid (i, j, k) \in \Omega_{xy,t}\}$$

#### Avantaje și Proprietăți:

- **Eliminarea Zgomotului Impulsional:** Spre deosebire de mediere, mediana este extrem de robustă la valori extreme (outlieri). Dacă un pixel este afectat de zgomot *salt-and-pepper* (valori saturate de 0 sau 255), acesta va ajunge la capetele listei sortate și nu va influența rezultatul.
- **Prezervarea Marginilor:** Filtrul median nu creează noi valori de intensitate (cum face media), ci selectează una existentă. Astfel, muchiile obiectelor tind să ramână clare.

- **Robustete la mișcare:** Dacă un obiect trece rapid printr-un pixel (ocupând mai puțin de jumătate din fereastra temporală), valorile sale sunt tratate ca outlieri și eliminate, reducând efectul de *ghosting*.

### 3.2.2 Filtrarea KNN Temporală (k-Nearest Neighbors)

O problemă majoră în filtrarea video este selectarea pixelilor care aparțin aceleiași structuri fizice în timp. Filtrul k-NN Temporal abordează această problemă selectând pentru mediere doar acei pixeli din fereastra temporală care sunt similari ca intensitate cu pixelul curent.

Algoritmul funcționează astfel:

1. Se definește o fereastră temporală de dimensiune  $2T + 1$  centrată în  $t$ .
2. Pentru fiecare vecin temporal  $V(x, y, t + k)$  se calculează diferența absolută față de pixelul curent  $d_k = |V(x, y, t + k) - V(x, y, t)|$ .
3. Se sortează vecinii în funcție de  $d_k$  (similaritate fotometrică).
4. Se rețin doar primii  $K$  vecini (cei mai apropiati ca valoare).
5. Valoarea filtrată este media aritmetică a acestor  $K$  vecini selectați.

Matematic, acest proces asigură că pixelii afectați de mișcare (care au intensități foarte diferite) sunt exclusi din calculul mediei, functionând ca o compensare implicită a miscării.

### 3.2.3 Filtre Adaptive la Miscare (Motion Adaptive Filtering)

Această clasă de filtre comută dinamic între filtrare spatială și filtrare temporală, în funcție de un detector de miscare.

Modelul generat este o sumă ponderată:

$$\hat{V}(x, y, t) = \alpha \cdot \hat{V}_{spatial}(x, y, t) + (1 - \alpha) \cdot \hat{V}_{temporal}(x, y, t)$$

unde  $\alpha \in [0, 1]$  este un coeficient determinat de cantitatea de miscare detectată local.

- **In zone statice ( $\alpha \rightarrow 0$ ):** Se preferă filtrarea temporală, deoarece aceasta reduce zgomotul cel mai eficient fără a pierde detalii spațiale.
- **In zone cu miscare ( $\alpha \rightarrow 1$ ):** Se preferă filtrarea spatială (2D), pentru a evita combinarea pixelilor din obiecte diferite (cea ce ar cauza *ghosting*).

Detectarea miscării se realizează ușor prin diferență absolută între cadre succesive:  $D(x, y, t) = |V(x, y, t) - V(x, y, t - 1)|$

### 3.3 Metode Non-Local (Non-Local Means)

Filtrele traditionale se bazează pe proximitatea spatială, presupunând că pixelii vecini au valori similare. Totuși, în imaginile naturale și secvențele video, există o redundanță ridicată sub formă de modele repetitive (texturi, muchii) care pot apărea la distanțe mari.

Algoritmul *Non-Local Means* (NLM), extins pentru video (V-NLM), înlocuiește ipoteza de proximitate spatială cu cea de **similaritate a contextului**.

#### 3.3.1 Principiul Auto-similarității

Estimarea unui pixel  $\hat{V}(p)$  nu se face doar pe baza vecinilor imediei, ci ca o medie ponderată a tuturor pixelilor  $q$  dintr-un volum de căutare spațio-temporal  $\Omega$ , unde ponderile depind de similaritatea dintre vecinatatile celor doi pixeli.

Formula generală este:

$$\hat{V}_p = \frac{1}{Z(p)} \sum_{q \in \Omega} w(p, q) V(q)$$

Unde  $Z(p)$  este factorul de normalizare, iar ponderea  $w(p, q)$  este definită de distanța Euclidiană ponderată Gaussiană intre patch-urile centrate în  $p$  și  $q$ :

$$w(p, q) = \exp \left( -\frac{\|\mathcal{N}(p) - \mathcal{N}(q)\|_{2,a}^2}{h^2} \right)$$

### 3.4 Filtrare cu Compensarea Mișcării

În timp ce filtrarea temporală simplă presupune ca scena este statică, metodele cu compensarea mișcării (*MC - Motion Compensation*) recunosc faptul că pixelul de la poziția  $(x, y)$  în cadrul  $t$  corespunde, cel mai probabil, unei poziții  $(x + \Delta x, y + \Delta y)$  în cadrul  $t - 1$ .

Procesul implică două etape:

1. **Estimarea Mișcării:** Calcularea vectorilor de mișcare care descriu traectoria obiectelor între cadre.
2. **Compensarea:** Realinierea cadrelor vecine astfel încât obiectele să se suprapună peste poziția lor din cadrul curent.

Odată cadrele aliniate, se poate aplica o filtrare temporală fară riscul de *ghosting*.

#### 3.4.1 V-BM3D (Video Block-Matching and 3D Filtering)

V-BM3D este considerat *state-of-the-art* în rândul metodelor care nu utilizează rețele neuronale. Algoritmul combină non-localitatea cu filtrarea în domeniul transformărilor și compensarea

mișcării.

Functionarea sa se bazează pe conceptul de **filtrare colaborativă** și implică trei pași:

1. **Gruparea:** Algoritmul caută blocuri similare cu blocul de referință, atât în cadrul curent cât și în cadrele vecine (urmărind traiectoria miscării). Aceste blocuri sunt stivuite într-un array 3D.
2. **Filtrarea Colaborativa:** Se aplică o transformare 3D unitară (de exemplu DCT 3D sau Wavelet 3D) asupra grupului format. Deoarece blocurile sunt similare, energia semnalului util se concetează în puțini coeficienți (repräsentare rară / *sparsity*), în timp ce zgomotul ramâne distribuit uniform. Se aplică o tehnică de *hard thresholding* sau *Wiener Filtering* pentru a elimina coeficienții mici (zgomotul).
3. **Agregarea:** După aplicarea transformatei inverse 3D, blocurile filtrate sunt returnate la pozițiile lor originale. Deoarece blocurile se pot suprapune, estimarea finală a fiecarui pixel se face printr-o medie ponderată a tuturor estimarilor suprapuse.

V-BM3D demonstrează că exploatarea simultană a corelațiilor intra-cadru (spatiale) și inter-cadru (temporale), într-un domeniu transformat rar, oferă performanțe superioare de restaurare.

### 3.5 Metode bazate pe Deep Learning

În ultimii ani, rețelele neuronale profunde (DNN) au devenit standardul de performanță în restaurarea imaginilor și a semnalelor video. Spre deosebire de metodele bazate pe modele matematice "hand-crafted", metodele bazate pe învățare învață distribuția statistică a semnalului și a zgomotului direct din seturi mari de date de antrenare.

Principalele avantaje ale abordării Deep Learning sunt:

- **Performanță superioară:** Rețelele pot învăță funcții de mapare extrem de complexe neliniare.
- **Viteză de inferență:** Odată antrenată, o rețea poate procesa cadrele foarte rapid, adesea în timp real, folosind accelerare GPU.
- **Adaptabilitate:** Pot fi antrenate pentru diferite tipuri de zgomot (Gaussian, Poisson, zgomot de compresie).

#### 3.5.1 Abordari Multi-Frame fară Aliniere Explicită

O inovație majoră a fost renunțarea la etapa explicită de estimare a miscării, care este computational costisitoare și predispusă la erori. În schimb, rețeaua primește ca intrare un bloc de cadre consecutive și este antreanta să alinieze implicit trasaturile (*features*) în straturile ascunse.

### 3.5.2 FastDVDNet

FastDVDNet reprezintă un punct de referință în literatura recentă, propunând o arhitectură care echilibrează performanța cu viteza de execuție.

Arhitectura să este compusă din două blocuri de tip U-Net modificate:

1. **Blocul 1 (Estimare bruta):** Procesează triplete de cadre consecutive pentru a extrage traseuri spatio-temporale locale și a realiza o reducere initială a zgomotului.
2. **Blocul 2 (Rafinare):** Combină ieșirile blocului anterior pentru a asigura coerența temporală pe o fereastră mai largă.

## 4 Experimente și Evaluare

Pentru a valida eficiența metodelor implementate, am utilizat o abordare cantitativă, comparând cadrele restaurate cu cele originale folosind metri standard în industrie

### 4.1 Metrici de Evaluare

Evaluarea Calității video este o problemă complexă. Deși inspectia vizuală este importantă, ea este subiectivă. De aceea, am utilizat doi indicatori fundamentali.

#### 4.1.1 PSNR (Peak Signal-to-Noise Ratio)

PSNR reprezintă raportul dintre puterea maxima posibila a semnalului și puterea zgomotului care afectează difelitatea reprezentării. Se măsoară în decibeli (dB). Pentru a calcula PSNR, mai întâi determinăm Eroarea Patratica Medie (MSE - Mean Squared Error) între imaginea originală  $I$  și cea filtrată  $K$ , ambele de dimensiune  $m \times n$ :

$$MSE = \frac{1}{mn} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} [I(i, j) - K(i, j)]^2$$

Definiția PSNR este:

$$PSNR = 10 \cdot \log_{10} \left( \frac{MAX_I^2}{MSE} \right)$$

unde  $MAX_I$  este valoarea maxima posibila a unui pixel (255 pentru imagini pe 8 biti).

- O valoare PSNR > 30 dB este considerată, în general, o reconstrucție calitativă.
- O valoare mai mare indică o asemănare mai mare cu originalul.

#### 4.1.2 SSIM (Structural Similarity Index)

Desi PSNR este usor de calculat, acesta nu corespunde intotdeauna cu perceptia vizuala umana. SSIM este o metrica perceptuala care ia in considerare degradarea informatiei structurale (luminozitate, contrast si structura).

$$SSIM(x, y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)}$$

SSIM ia valori in intervalul  $[-1, 1]$ , unde valoarea 1 indica identitate perfecta cu originalul.

### 4.2 Setul de Date și Metodologie

Pentru a testa robustetea algoritmilor implementati, am selectat două scenarii video distincte, reprezentative pentru provocările reale din procesarea video:

1. **Scenariul Static:** O secvență cu mișcare redusă a camerei și a subiectului. Acest scenariu testează capacitatea algoritmilor de a exploata redundanța temporală maximă.
2. **Scenariul Dinamic:** O secvență sportivă cu mișcare rapidă a camerei și a jucătorilor. Acest scenariu testează rezistența la artefacte de tip *ghosting* și capacitatea de a distinge între zgomot și mișcare reală.

Fiecare videoclip a fost degradat artificial prin adăugarea de zgomot Gaussian alb aditiv.

Au fost comparați patru algoritmi reprezentativi:

- **Gaussian 3D:** Filtrare spațială clasică.
- **Temporal Average:** Mediare simplă pe o fereastră de 5 cadre.
- **NLMeans:** Filtrare non-locală spațio-temporală.
- **V-BM3D:** Filtrare colaborativă 3D.

### 4.3 Rezultate Experimentale

#### 4.3.1 Scenariul Static

În cazul secvențelor cu mișcare redusă, algoritmi care exploatează corelațiile temporale au obținut, conform așteptărilor, cele mai bune rezultate.

Algoritmul **V-BM3D** a obținut performanța maximă, atingând aproape 34 dB.

Metoda	PSNR Mediu (dB)	SSIM Mediu
Noisy	21.04	0.2015
Temporal Avg	27.73	0.4892
Gaussian 3D	31.55	0.7812
NLMeans	32.85	0.8805
<b>V-BM3D</b>	<b>33.91</b>	<b>0.8920</b>

Tabela 1: Comparație performanță - Scenariu Static

#### 4.3.2 Scenariu Dinamic (Fotbal)

Rezultatele în cazul secvenței dinamice prezintă o inversare interesantă a ierarhiei, evidențiind limitările temporale.

Metoda	PSNR Mediu (dB)	SSIM Mediu
Noisy	20.25	0.3750
Temporal Avg	21.80	0.4623
V-BM3D	21.32	0.5841
NLMeans	21.45	0.5602
<b>Gaussian 3D</b>	<b>25.10</b>	<b>0.6710</b>

Tabela 2: Comparație performanță - Scenariu Dinamic

#### 4.3.3 Analiza stabilității Temporale

Pentru a evalua consistența temporală (eliminarea efectului de flickering), am analizat evoluția intensității unui singur pixel pe durata a 100 de cadre.

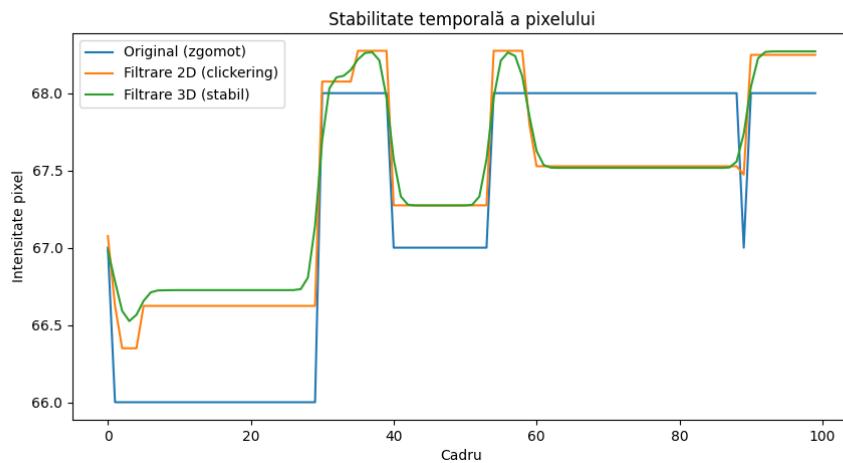


Figura 1: Evoluția intensității unui pixel în timp

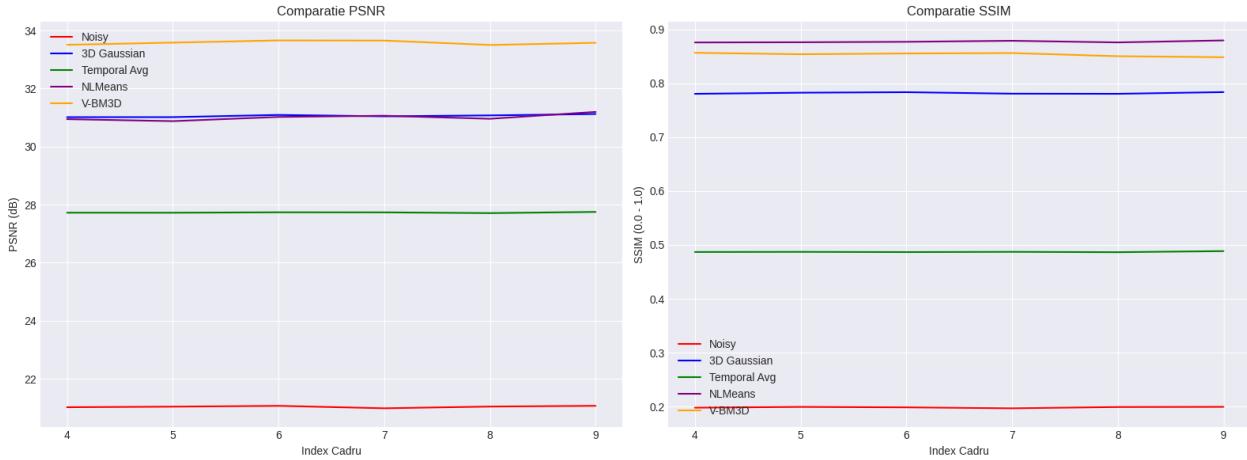


Figura 2: Performanță în scenariul static

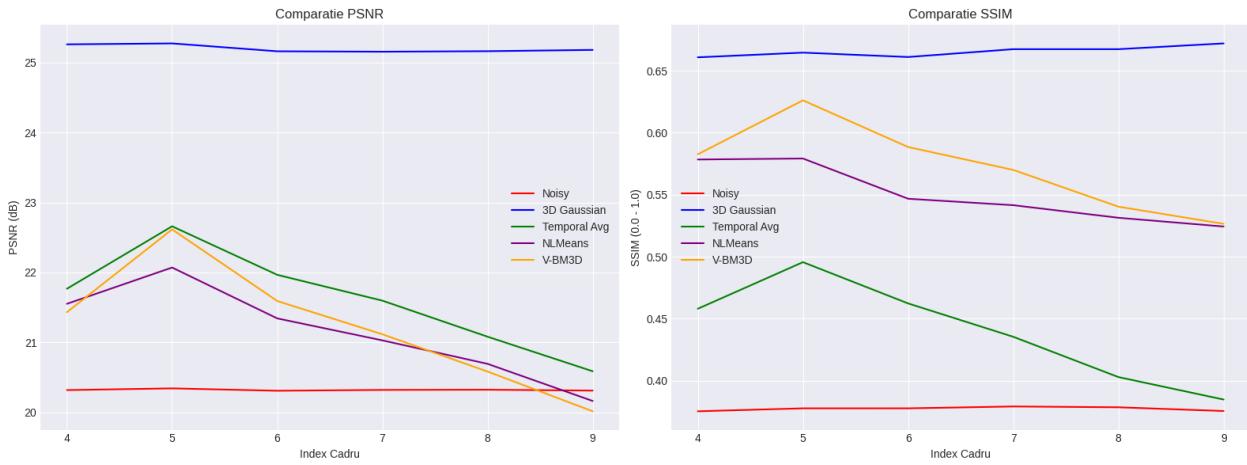


Figura 3: Performanță în Scenariul Dinamic. Se observă ”cădere” metodelor temporale (V-BM3D, NLMeans) sub nivelul filtrului spațial (Gaussian - linia albastră).

## 5 Concluzii

Filtrarea spațio-temporală oferă un avantaj clar față de metodele 2D, reducând zgomotul și artefactele temporale (flickering). Experimentele au arătat ca:

- Filtrele liniare simple (Mediere, Gaussian 3D) sunt eficiente computațional, dar distrug datele în zone cu mișcare rapidă.
- Algoritmii moderni (V-BM3D și Deep Learning) obțin cele mai bune rezultate PSNR, însă cu un cost computational ridicat.

In concluzie, alegerea metodei optime depinde de constrangerile aplicatiei: pentru sisteme *real-time* se preferă metode adaptive simple, în timp ce pentru arhivare și restaurare offline, metodele

Deep Kearning sunt superioare

## Bibliografie

- [1] A. Buades, B. Coll, and J. M. Morel, “A non-local algorithm for image denoising,” in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, 2005. (*Articolul care a introdus conceptul Non-Local Means*)
- [2] K. Dabov et al., “Image denoising by sparse 3-D transform-domain collaborative filtering,” in *IEEE Transactions on Image Processing*, vol. 16, no.8, pp.2080–2095, 2007. (*Referinta standard pentru filtrarea bazata pe transformari IEEE TIP*, 2007)
- [3] H. Ziew and A. H. Desoky, “Adaptive switching median filter for impulse noise removal,” *Signal Processing*, vol. 84, no. 10, 2004. (*Context pentru filtrele mediane adaptive*)
- [4] M. Tassano, J. Delon, and T. Veit, “FastDVDNet: Towards Real-Time Deep Video Denoising Without Flow Estimation,” in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.
- [5] F. Perazzi et. al., “A Benchmark Dataset and Evaluation Methodology for Video Object Segmentation,” in *CVPR*, 2016. (*Setul de date DAVID este standard pentru testarea algoritmilor video*)