

ESTRAZIONE DEL PRNU TRAMITE DENOISER BASATO SU RETI NEURALI CONVOLUZIONALI (DRUNET)

Autori: **Mirco Ceccarelli, Francesco Argentieri**

Relatori: **Prof. Alessandro Piva, Dott. Dasara Shullani, Dott. Daniele Baracchi**

1 Introduzione

In questo elaborato è stato preso in esame l'ambito della Multimedia Forensics, il quale studia le tracce intrinseche e artefatti caratteristici che vengono lasciati su un qualsiasi media digitale (immagini o video) durante la loro creazione o altri processi a cui sono sottoposti.

In particolare il focus è stato posto sulla disciplina della *Device Identification*, il cui obiettivo è risalire a quale dispositivo ha acquisito un determinato contenuto multimediale.

Per fare ciò viene sfruttato un rumore che viene introdotto dai sensori di ogni fotocamera digitale: il **PRNU** (Photo Response Non Uniformity), che è distintivo di ogni singolo *device*.

Partendo dunque da un'immagine reale è necessario inizialmente applicarle un filtro di denoising per ottenere un'immagine priva di rumore. Dopodiché viene calcolato il rumore residuo facendo la differenza tra queste due immagini. Vengono svolti questi passaggi perché è più facile stimare il PRNU dal rumore residuo che non dall'immagine di partenza. Infine il PRNU del dispositivo viene confrontato con il rumore residuo dell'immagine di prova.

Di seguito viene riportato un breve schema del processo di *Device identification* tramite PRNU:

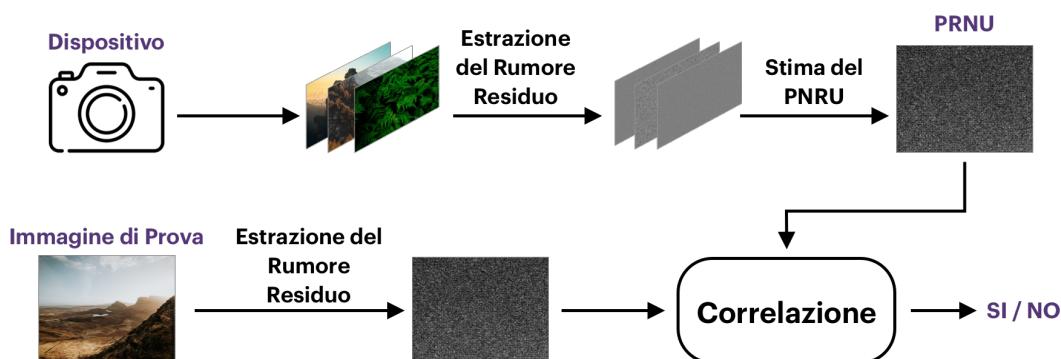


Figura 1: PRNU: Device Identification

2 Attività di Ricerca

Questo progetto ha come obiettivo il verificare le performance del sistema di *Device Identification* sostituendo la fase di denoising finora basata su modelli, con un denoiser basato su reti neurali convoluzionali.

Per la fase di denoising sono state varate due possibilità:

- **DRUNet denoiser prior**: rete in grado di trattare vari livelli di rumore attraverso un unico modello [1].

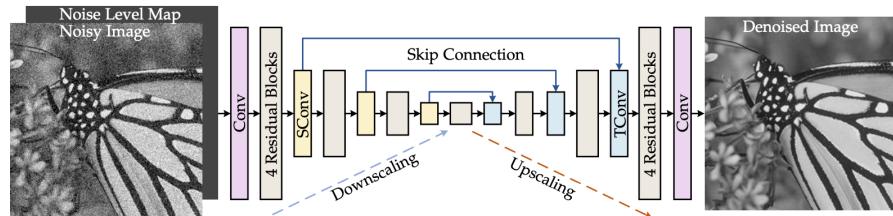


Figura 2: Architettura di DRUNet

- **RIDNet**: rete single-stage che sfrutta la *Feature Attention* per il denoising delle immagini [2].

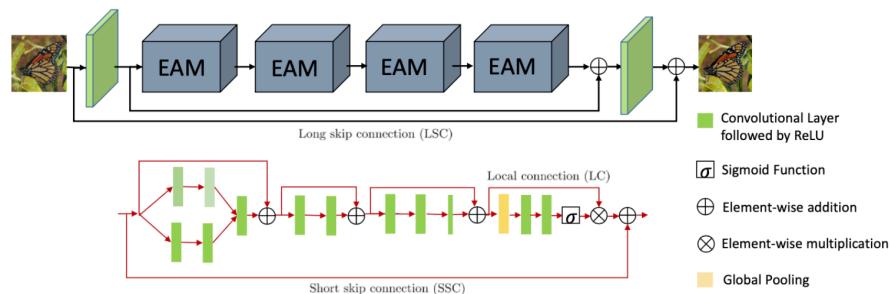


Figura 3: Architettura di DRUNet

A seguito di un approfondimento di queste due reti neurali convoluzionali, la scelta è ricaduta su **DRUNet**, in quanto forniva ottime prestazioni in base ai risultati degli esperimenti riportati nel paper [1].

3 Dettagli dell'Implementazione

La *baseline* di questo elaborato è stato il lavoro svolto da parte di studenti del Politecnico di Milano per l'estrazione del PRNU sviluppato in Python [3].

Nel loro progetto la funzione che si occupa dell'estrazione del rumore residuo è un algoritmo *model-based* (**NoiseExtract**).

Un primo passo quindi è stato quello di sostituirlo con un algoritmo *leaning-based*, ovvero **DRUNet**.

Lo step successivo è stato quello di modificare i livelli di rumore σ con cui la rete veniva testata (con $\sigma = 15, 50, 100$); infatti per avere un addestramento supervisionato della rete veniva aggiunto un rumore gaussiano bianco additivo (AWGN) di livello σ .

Infine dopo aver eseguito tutti i test con il dataset fornito dagli studenti del Politecnico di Milano, sono stati ripetuti e ampliati gli stessi esperimenti sul dataset *reVISION_dataset_base* del laboratorio **LESC** dell'Università degli Studi di Firenze.

Di seguito saranno illustrati i risultati degli esperimenti svolti.

4 Risultati degli Esperimenti

Tutti i risultati sono stati ottenuti in termini di **AUC** (Area Under Curve), utilizzando come misura di correlazione la **PCE** (Peak to Correlation Energy).

4.1 Esperimenti con Dataset Polimi

Il dataset utilizzato in questo *batch* di esperimenti è costituito da immagini acquisite da **6 dispositivi** diversi, in particolare sono state utilizzate:

- **60 immagini flat** per il calcolo della fingerprint (10 per ogni dispositivo);
- **20 immagini naturali totali** per i test.

4.1.1 Risultati con NoiseExtract

Il valore di **AUC sulla PCE** in questo caso è pari a **0.81**. Per verificare l'efficacia dell'algoritmo, avendo un numero ridotto di immagini su cui testarlo, è stato possibile raffigurare i valori della PCE su degli histogrammi (uno per ogni dispositivo presente nel dataset).

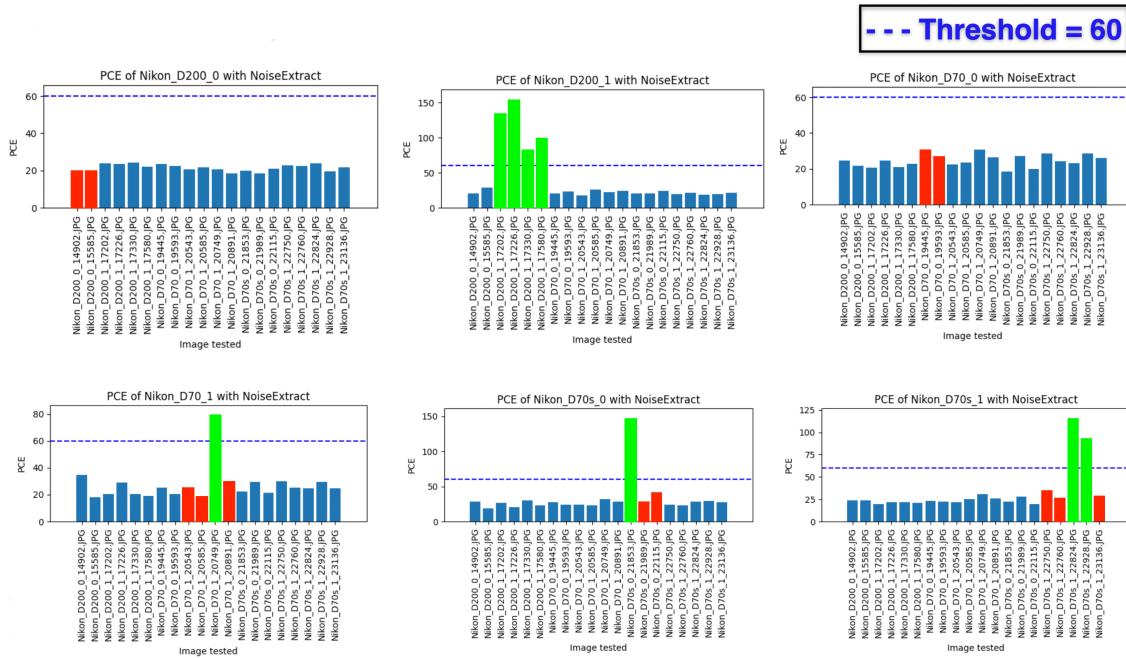


Figura 4: Istogrammi con metodo NoiseExtract

In particolare sono evidenziati in **verde** le barre relative alle immagini scattate dallo stesso dispositivo preso in esame in quel particolare istogramma, la cui PCE è superiore alla soglia di 60. In **rosso** invece sono evidenziate le barre relative alle immagini scattate dallo stesso dispositivo preso in esame in quel particolare istogramma, la cui PCE è inferiore alla soglia di 60.

4.1.2 Risultati con DRUNet15

Con il termine **DRUNet15** si intende il metodo DRUNet con livello di rumore $\sigma = 15$. Il valore di **AUC sulla PCE** in questo caso è pari a **0.70**. Abbiamo i seguenti histogrammi:

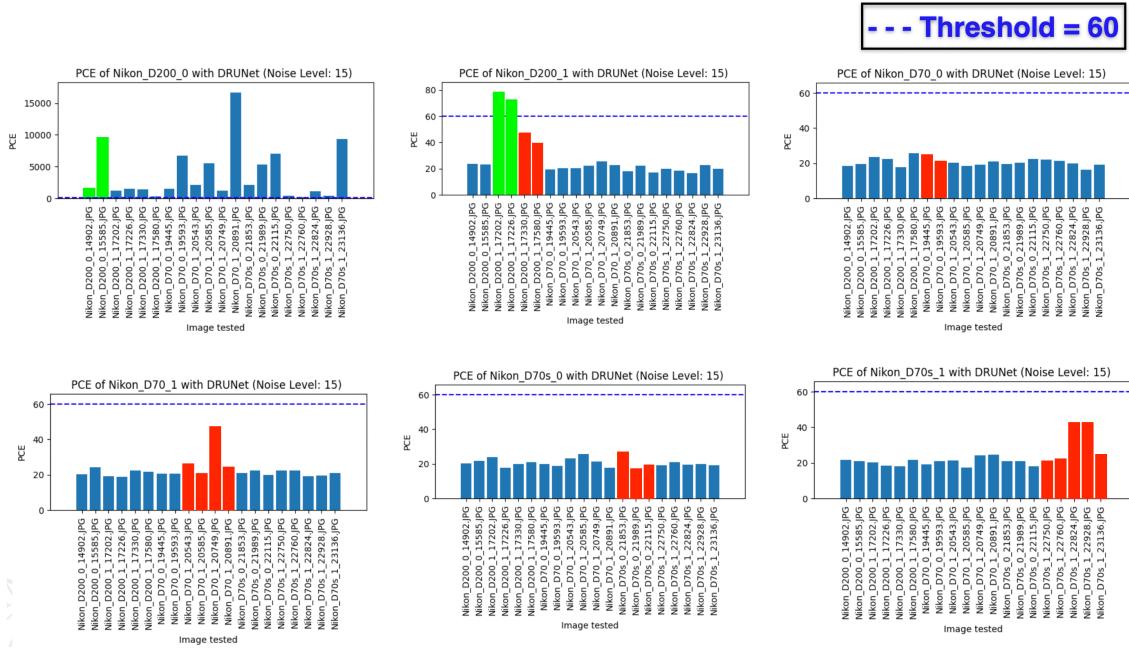


Figura 5: Istogrammi con metodo DRUNet15

4.1.3 Risultati con DRUNet50

Con il termine **DRUNet50** si intende il metodo DRUNet con livello di rumore $\sigma = 50$. Il valore di **AUC sulla PCE** in questo caso è pari a **0.82**. Abbiamo i seguenti histogrammi:

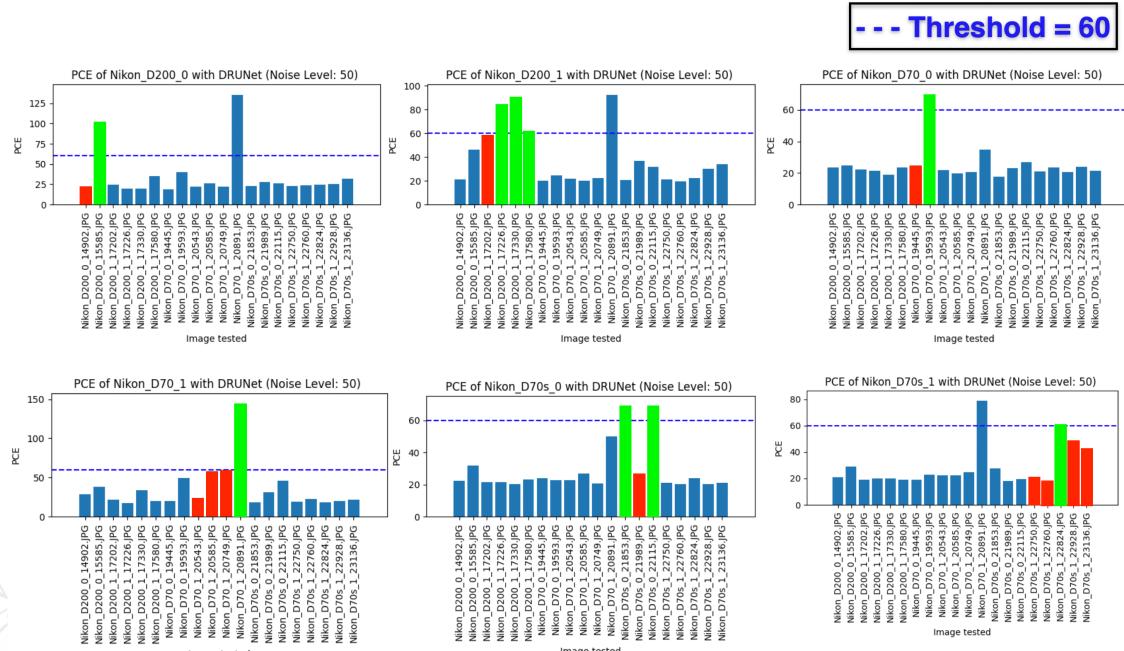


Figura 6: Istogrammi con metodo DRUNet50

4.1.4 Confronto sul dataset del Polimi

Viene riportato adesso il confronto in termini di AUC sulla PCE tra i metodi sopra analizzati: **NoiseExtract**, **DRUNet15**, **DRUNet50**. Per semplicità sono riassunti nella seguente tabella:

	NoiseExtract	DRUNet15	DRUNet50
AUC	0.81	0.70	0.82

Tabella 1: Confronto risultati con dataset Polimi

4.2 Esperimenti con Dataset reVISION_dataset_base

Il dataset utilizzato in questo *batch* di esperimenti è costituito da immagini acquisite da **32 dispositivi** diversi, in particolare sono state utilizzate:

- **1600 immagini flat** per il calcolo della fingerprint (50 per ogni dispositivo);
- **640 immagini naturali** per i test (20 per ogni dispositivo).

In questo insieme di esperimenti, avendo un numero più elevato di immagini a disposizione, vengono mostrate le curve **ROC** (Receiver Operating Characteristic) e dei **grafici** che mostrano la distribuzione delle seguenti statistiche:

- True Positive (**TP**) (con PCE > 60)
- False Positive (**FP**) (con PCE > 60)
- True Negative (**TN**) (con PCE < 60)
- False Negative (**FN**) (con PCE < 60)
- True Positive Rate (**TPR**) dove $TPR = \frac{TP}{TP+FN}$
- False Positive Rate (**FPR**) dove $FPR = \frac{FP}{FP+TN}$

I dispositivi sono così suddivisi:

- **Samsung**: D01, D08, D11, D22, D26, D27, D31, D35
- **Apple**: D02, D05, D06, D09, D10, D13, D14, D15, D18, D19, D20, D29, D34
- **Sony**: D12
- **Huawei**: D16, D28, D30, D33
- **Microsoft**: D17
- **Wiko**: D21
- **Asus**: D23
- **Xiaomi**: D24
- **OnePlus**: D25, D32

4.2.1 Risultati con NoiseExtract

Il valore di **AUC sulla PCE** in questo caso è **0.97**. La curva ROC ottenuta è la seguente:

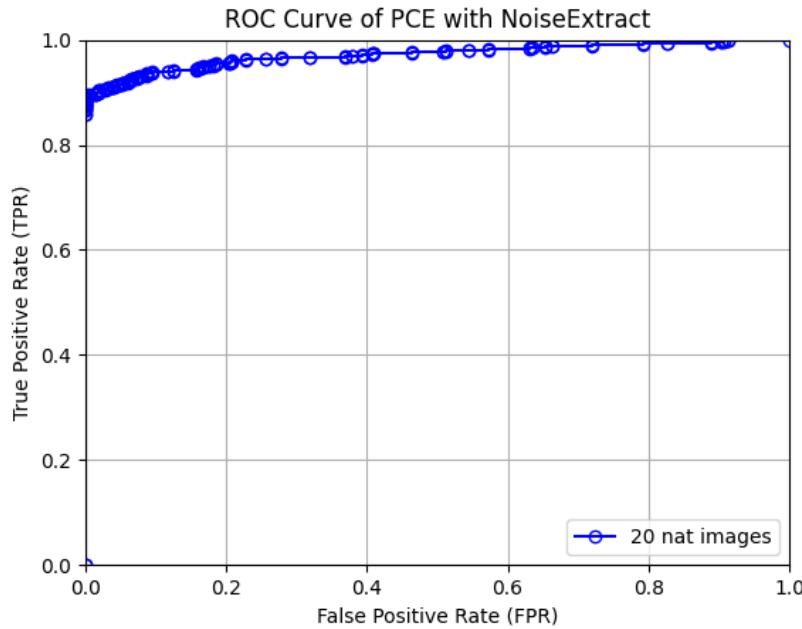


Figura 7: Curva ROC della PCE con il metodo NoiseExtract

Sono stati inoltre prodotti i seguenti grafici e tabelle riassuntive:

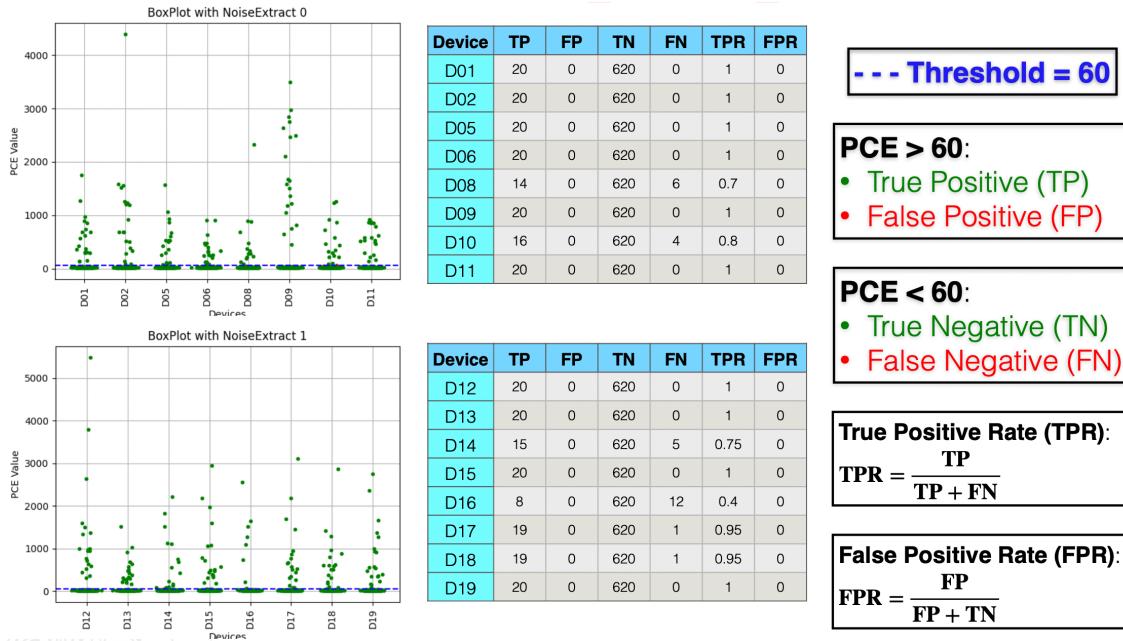


Figura 8: Parte 1/2

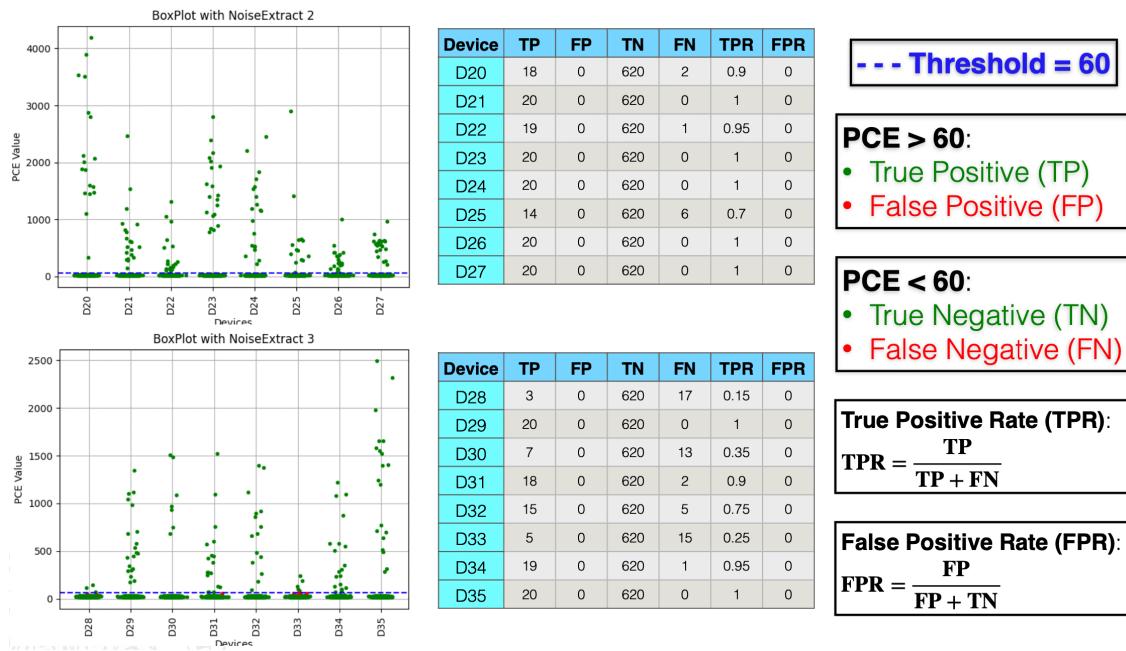


Figura 9: Parte 2/2

4.2.2 Risultati con DRUNet15

Con il termine **DRUNet15** si intende il metodo DRUNet con livello di rumore $\sigma = 15$. Il valore di **AUC sulla PCE** in questo caso è **0.56**. La curva ROC ottenuta è la seguente:

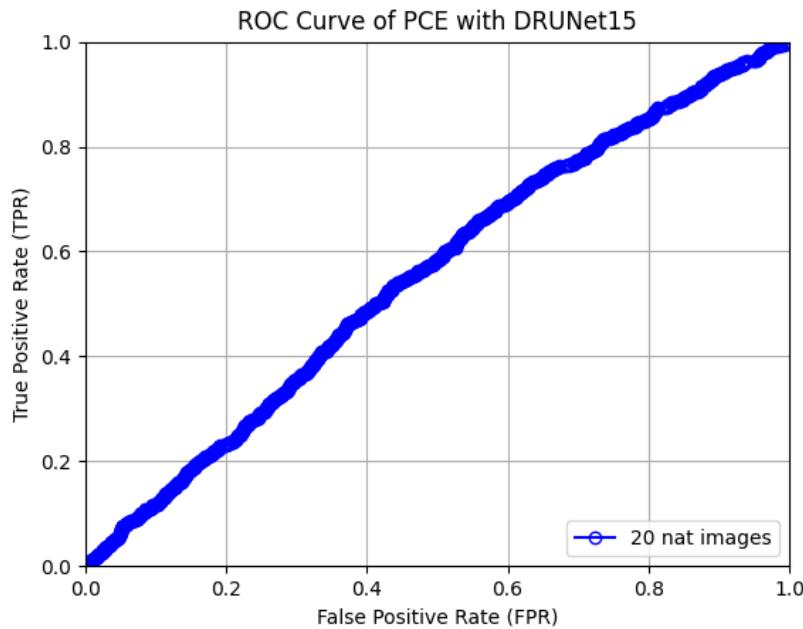


Figura 10: Curva ROC della PCE con il metodo DRUNet15

Sono stati inoltre prodotti i seguenti grafici e tabelle riassuntive:

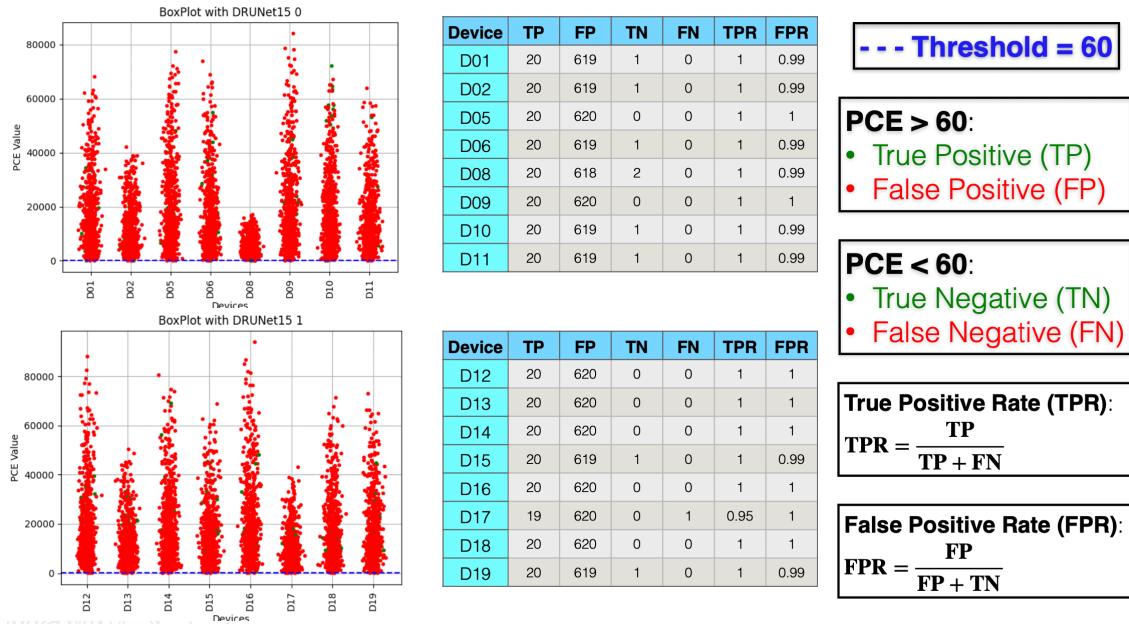


Figura 11: Parte 1/2

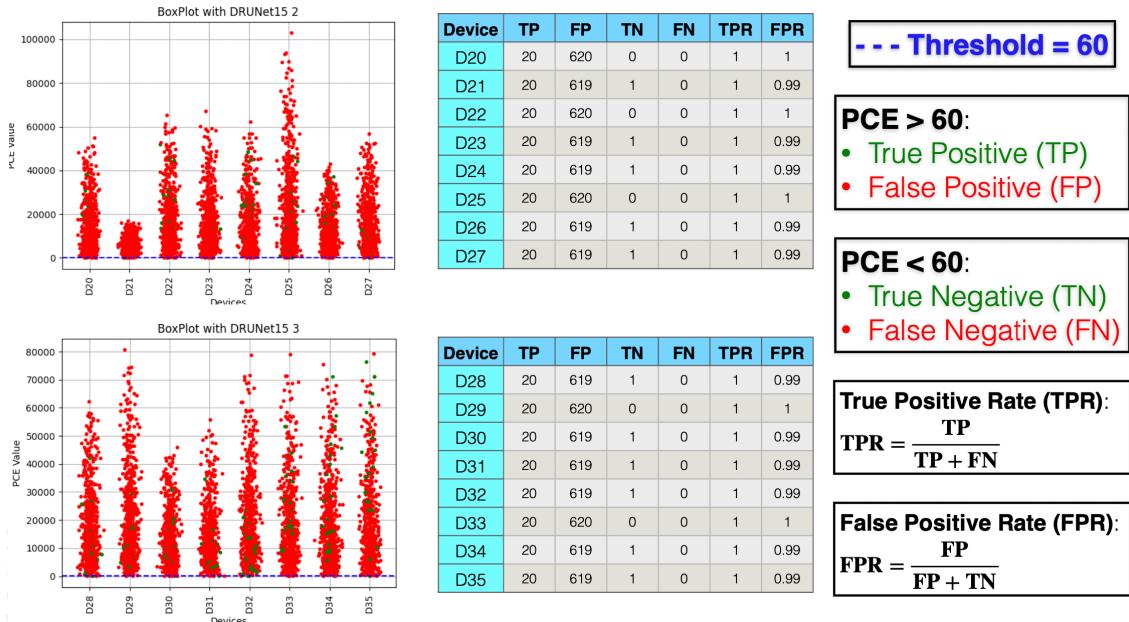


Figura 12: Parte 2/2

Con **DRUNet15** si può dunque vedere un netto peggioramento nelle prestazioni rispetto al caso precedente. Infatti analizzando la curva ROC è praticamente la diagonale del grafico, il che significa che è paragonabile a un classificatore casuale.

4.2.3 Risultati con DRUNet50

Con il termine **DRUNet50** si intende il metodo DRUNet con livello di rumore $\sigma = 50$. Il valore di **AUC sulla PCE** in questo caso è **0.83**. La curva ROC ottenuta è la seguente:

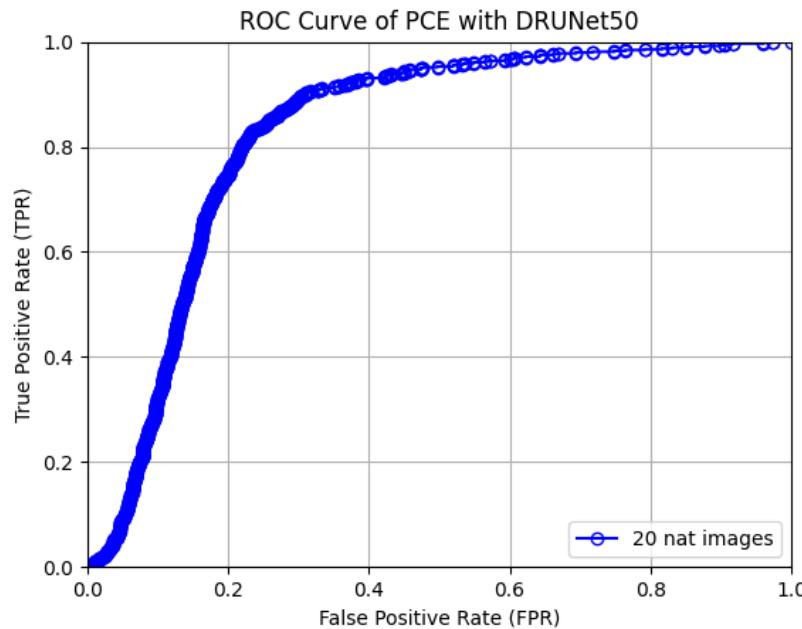


Figura 13: Curva ROC della PCE con il metodo DRUNet50

Sono stati inoltre prodotti i seguenti grafici e tabelle riassuntive:

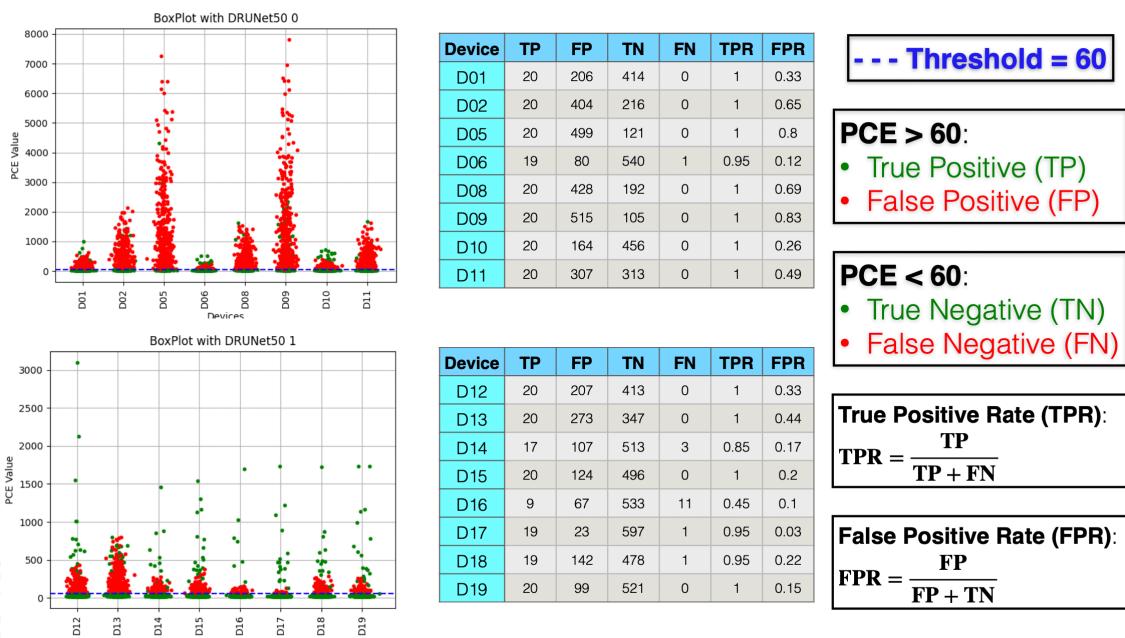


Figura 14: Parte 1/2

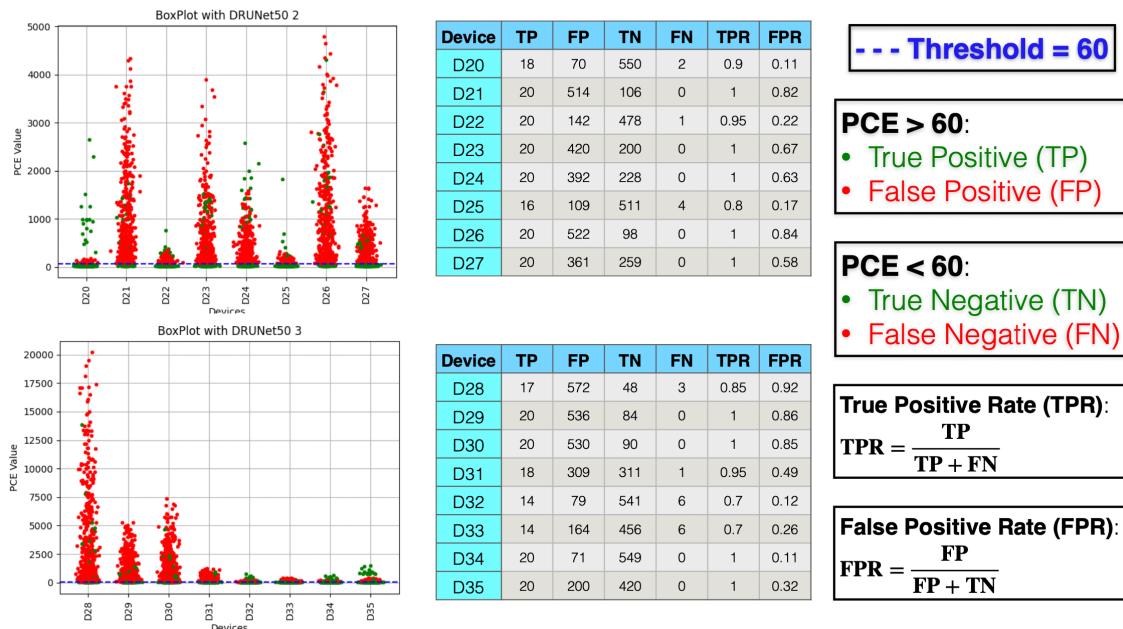


Figura 15: Parte 2/2

DRUNet50, come è possibile vedere dalla curva ROC, ha prestazioni paragonabili a quelle ottenute con il metodo **NoiseExtract**, con il vantaggio però di non essere vincolato ad un modello, essendo DRUNet50 un metodo *learning-based*.

Di seguito sono inoltre illustrate le curve ROC e relativo valore AUC, divise per singolo dispositivo:

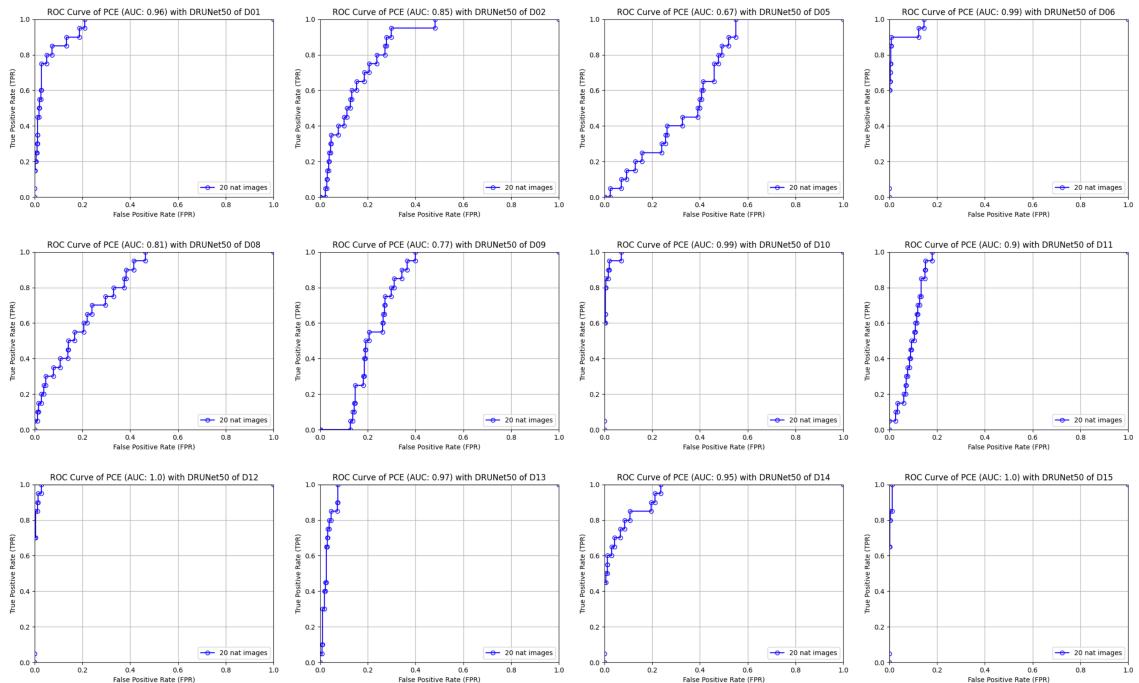


Figura 16: Parte 1/3

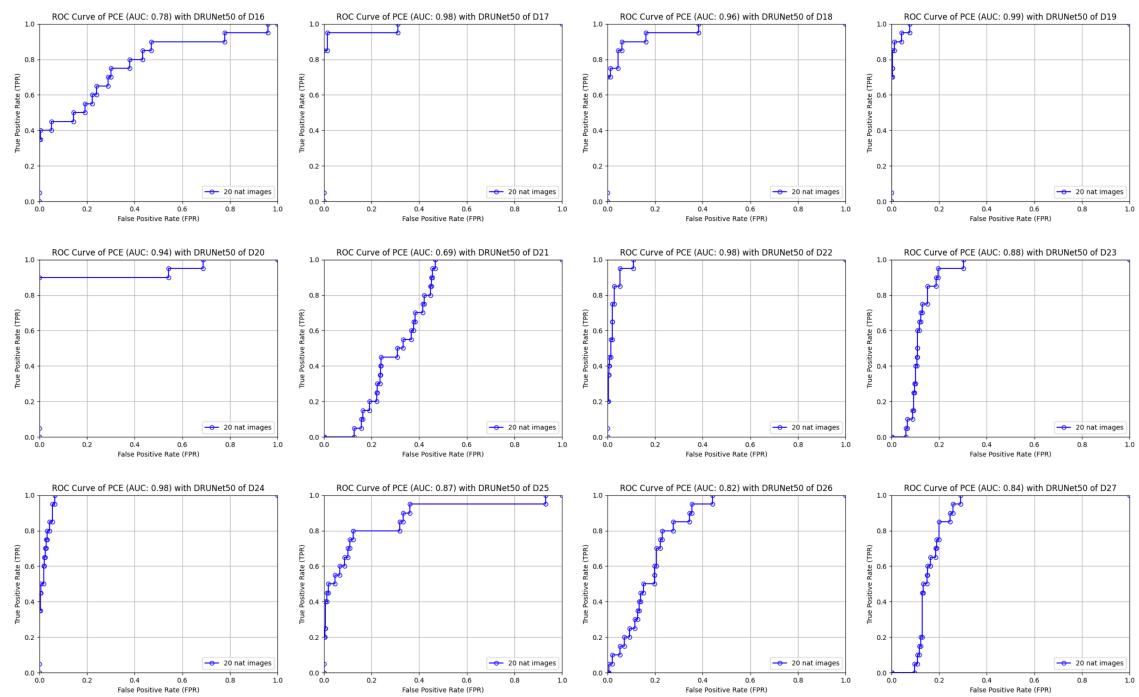


Figura 17: Parte 2/3

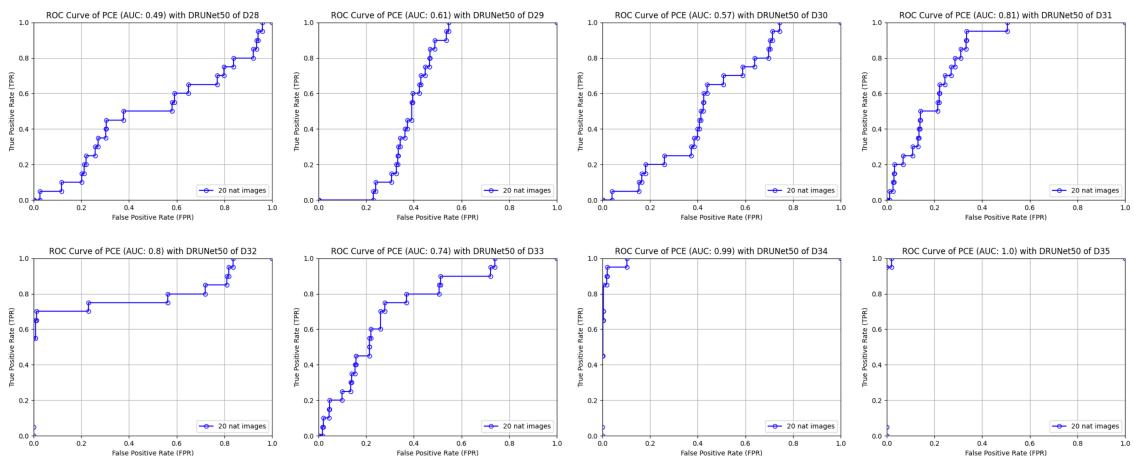


Figura 18: Parte 3/3

Da queste curve ROC è possibile osservare che i dispositivi **Huawei**, con una $AUC < 0.75$, abbiano performance peggiori rispetto agli altri device.

4.2.4 Risultati con DRUNet100

Con il termine **DRUNet100** si intende il metodo DRUNet con livello di rumore $\sigma = 100$. Il valore di **AUC sulla PCE** in questo caso è **0.90**. La curva ROC ottenuta è la seguente:

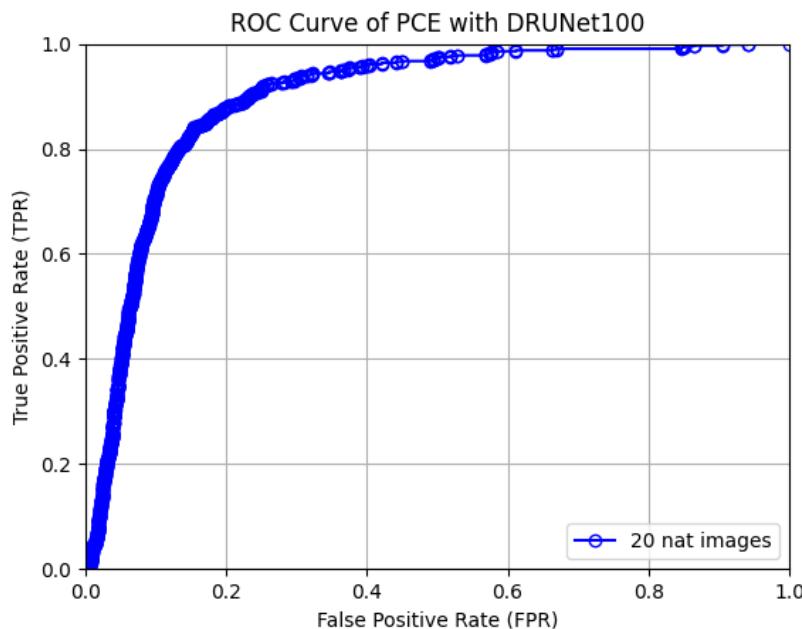


Figura 19: Curva ROC della PCE con il metodo DRUNet100

Sono stati inoltre prodotti i seguenti grafici e tabelle riassuntive:

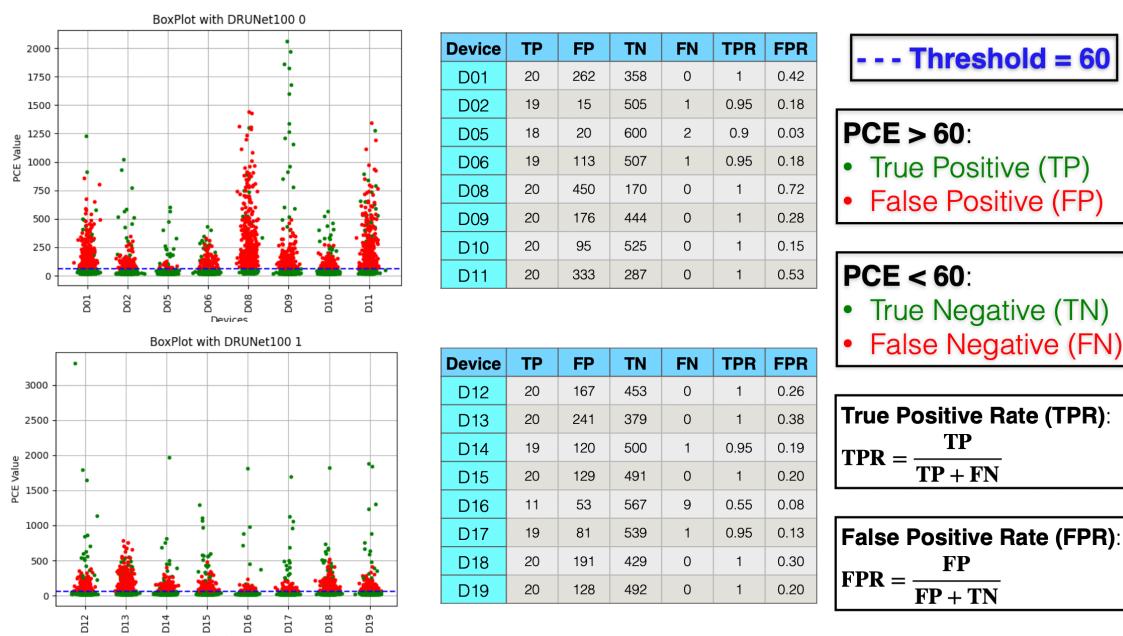


Figura 20: Parte 1/2

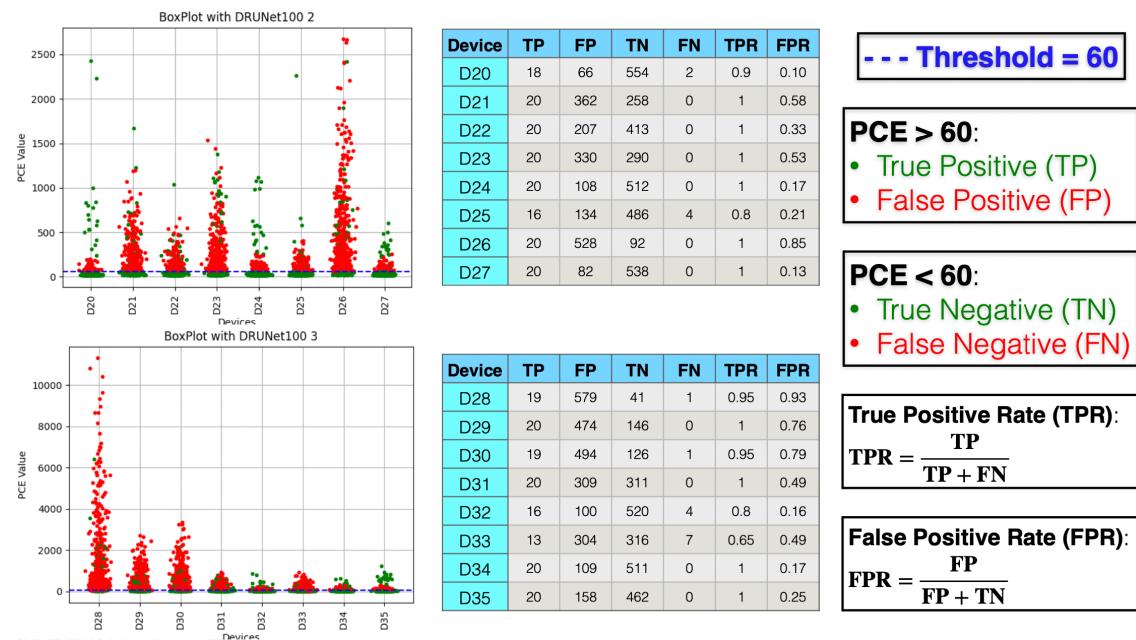


Figura 21: Parte 2/2

Di seguito sono inoltre illustrate le curve ROC e relativo valore AUC, divise per singolo dispositivo:

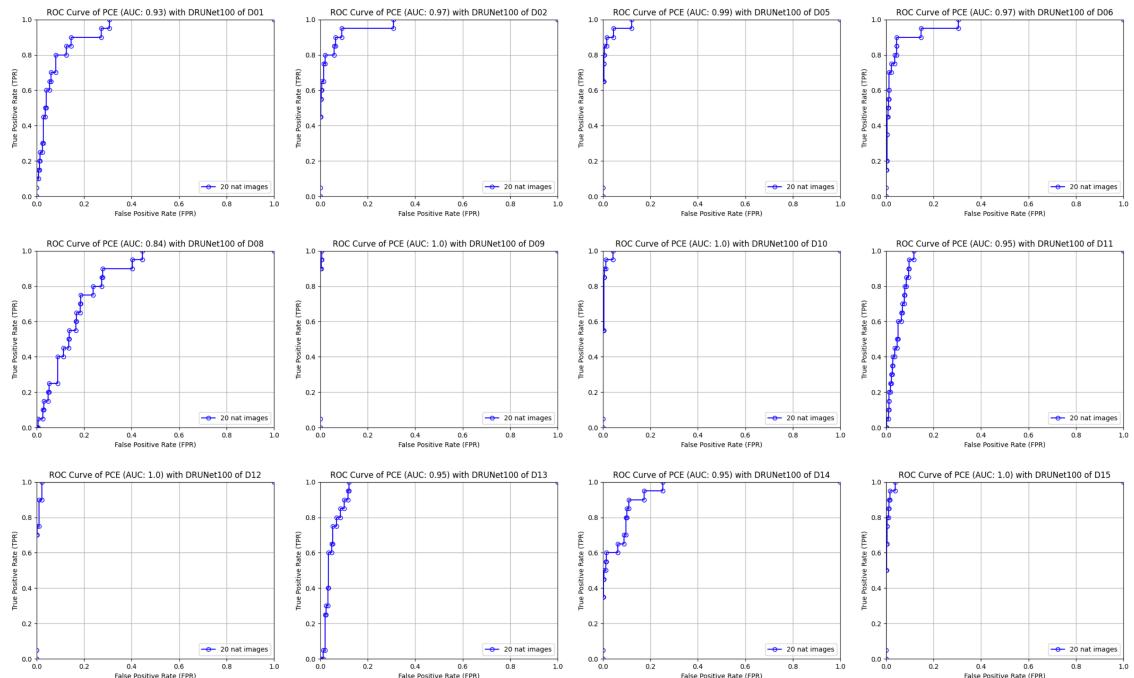


Figura 22: Parte 1/3

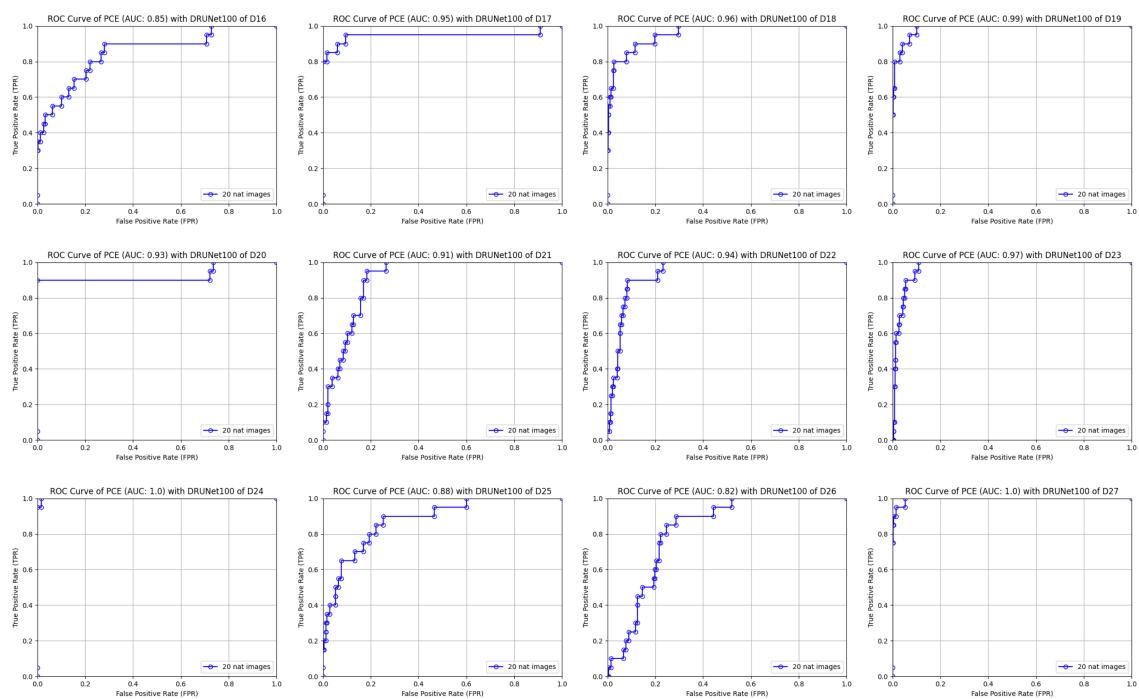


Figura 23: Parte 2/3

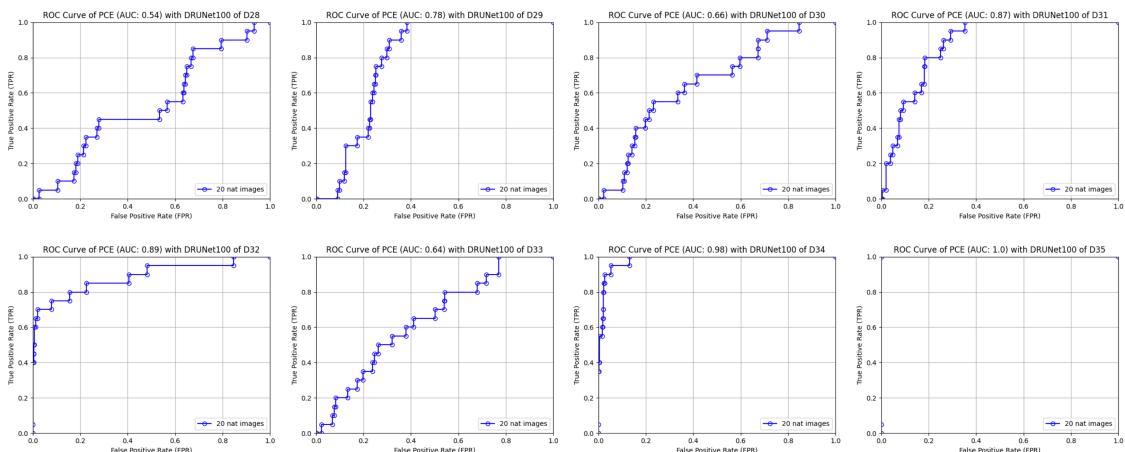


Figura 24: Parte 3/3

Come nel caso con DRUNet50, anche con **DRUNet100** i dispositivi che mostrano le performance peggiori in termini di AUC della PCE sono gli **Huawei**.

Sono stati inoltre calcolati i seguenti tempi di elaborazione: estrazione del rumore residuo per immagine (**5 secondi**) e calcolo della fingerprint per dispositivo (**4,5 minuti**).

4.2.5 Confronto sul dataset *reVISION_dataset_base*

Viene riportato adesso il confronto in termini di AUC sulla PCE tra i metodi sopra analizzati: **NoiseExtract**, **DRUNet15**, **DRUNet50** e **DRUNet100**. Per semplicità sono riassunti nella seguente tabella:

	NoiseExtract	DRUNet15	DRUNet50	DRUNet100
AUC	0.97	0.56	0.83	0.90

Tabella 2: Confronto risultati con dataset *reVISION_dataset_base*

Per confrontare l'andamento delle curve ROC ottenute con i 4 metodi sopra proposti, si ha di seguito il seguente grafico comparativo:

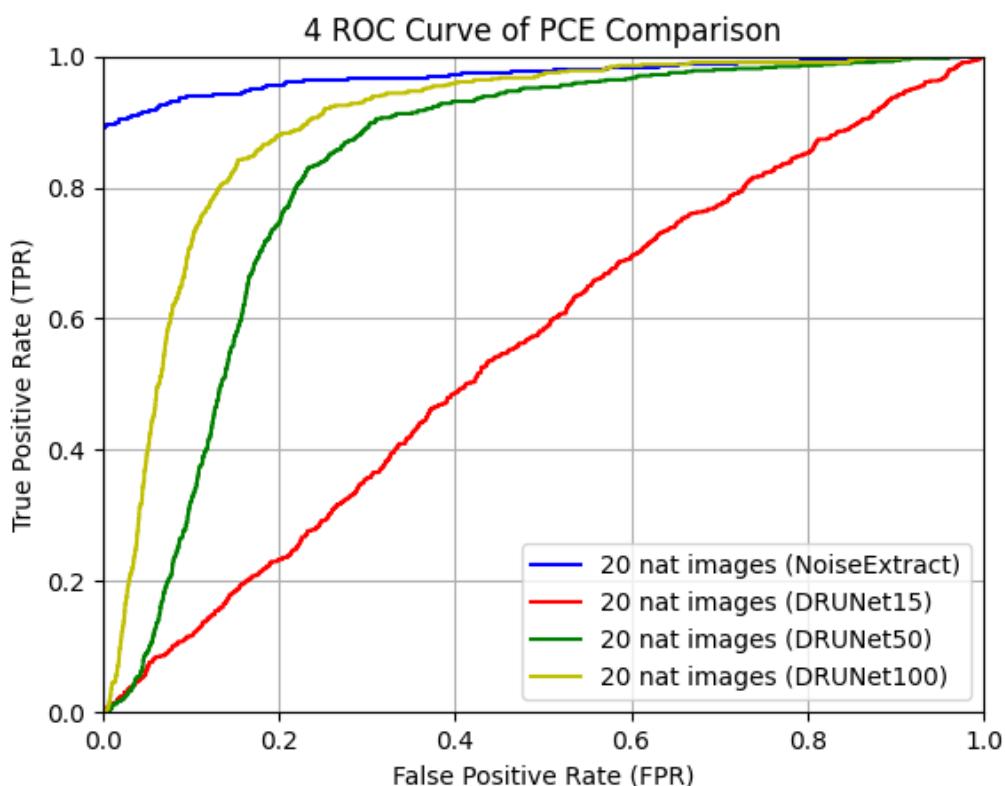


Figura 25: Confronto delle 4 curve ROC

5 Ablation Studies

Per quanto riguarda il modello con **DRUNet50**, abbiamo **rimosso delle operazioni** per osservare come si comportava il sistema complessivo in termini di prestazioni.

In particolare abbiamo rimosso:

- Solo l'operazione di **Zero Mean**
- Solo l'operazione di **Wiener DFT**
- Entrambe le operazioni di **Zero Mean** e di **Wiener DFT**

Gli esperimenti sono sempre stati effettuati sul dataset **reVISION_dataset_base**.

5.1 Rimozione dell'operazione di Zero Mean

Con il termine **DRUNet50NoZeroMean** si intende il metodo DRUNet50 a cui è stata rimossa l'operazione **zero_mean()**. Il valore di **AUC sulla PCE** in questo caso è **0.83**. La curva ROC ottenuta è la seguente:

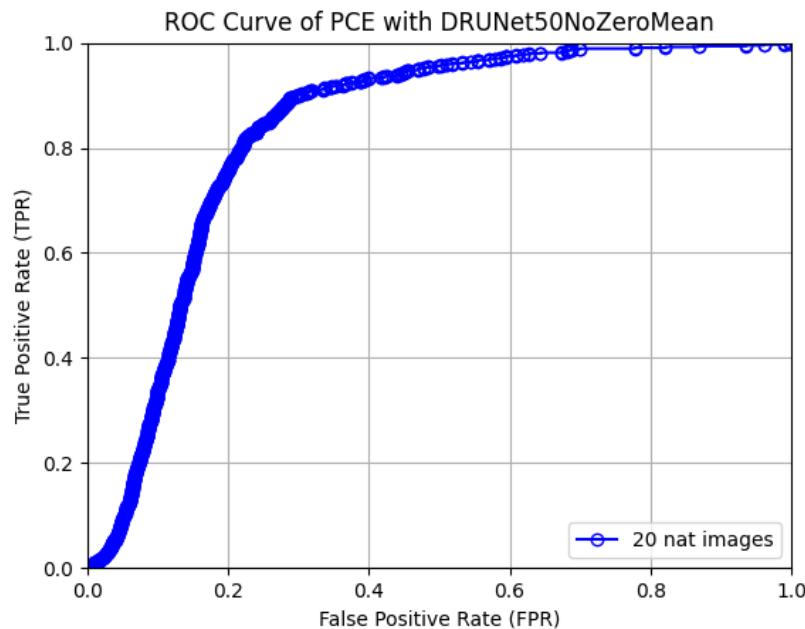


Figura 26: Curva ROC della PCE con il metodo DRUNet50NoZeroMean

Sono stati inoltre prodotti i seguenti grafici e tabelle riassuntive:

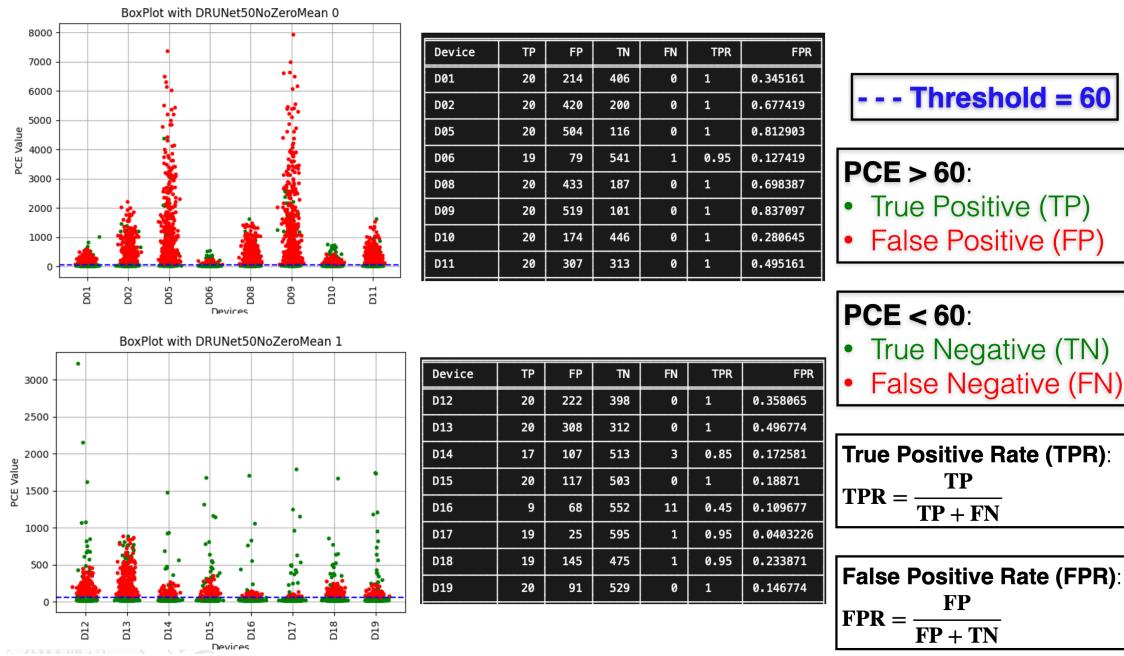


Figura 27: Parte 1/2

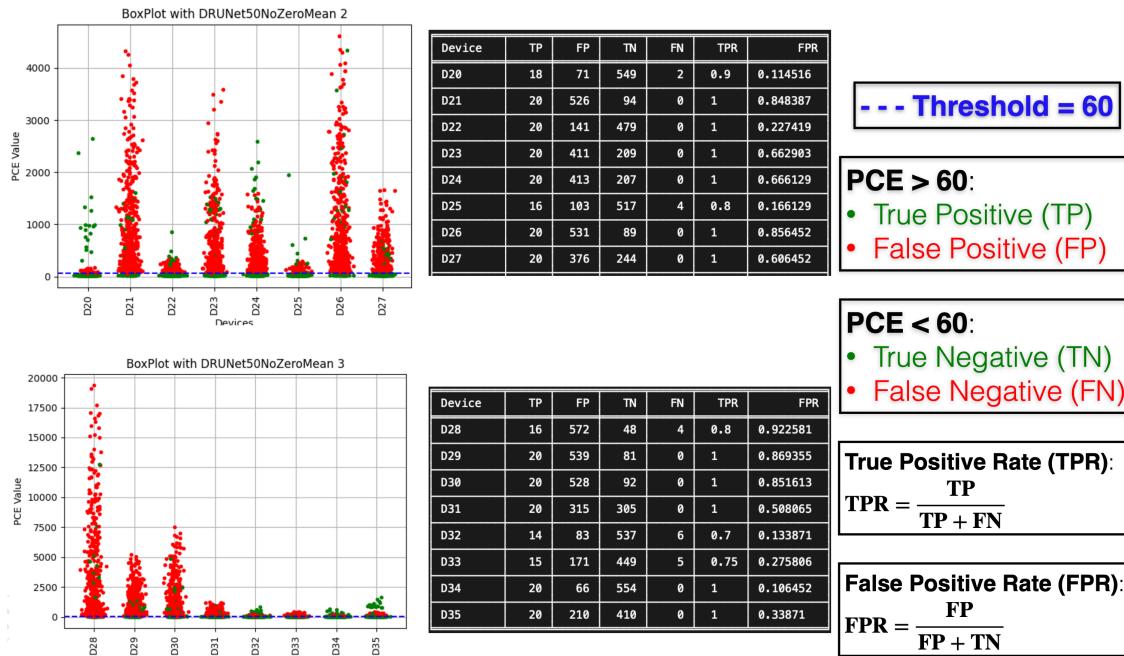


Figura 28: Parte 2/2

5.2 Rimozione dell'operazione di Wiener DFT

Con il termine **DRUNet50NoWiener** si intende il metodo DRUNet50 a cui è stata rimossa l'operazione ***wiener_dft()***. Il valore di **AUC sulla PCE** in questo caso è **0.62**. La curva ROC ottenuta è la seguente:

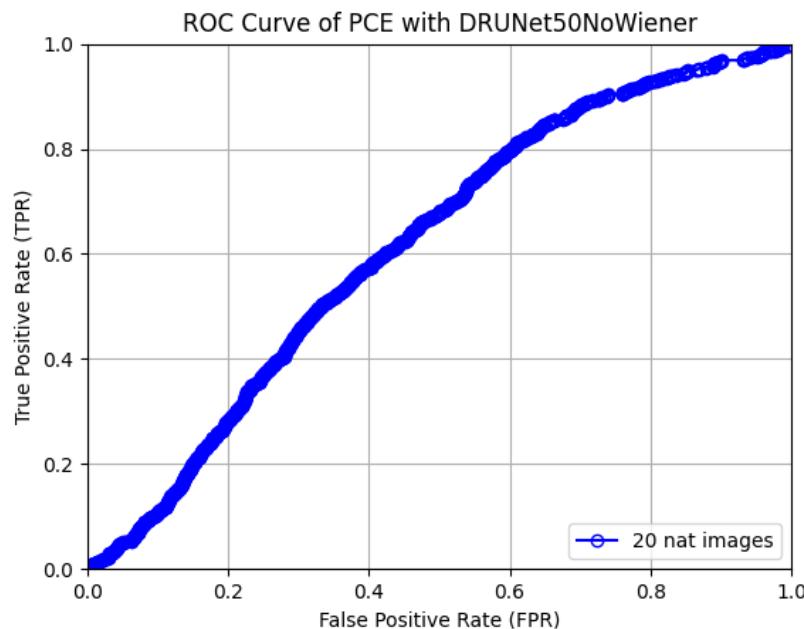


Figura 29: Curva ROC della PCE con il metodo DRUNet50NoWiener

Sono stati inoltre prodotti i seguenti grafici e tabelle riassuntive:

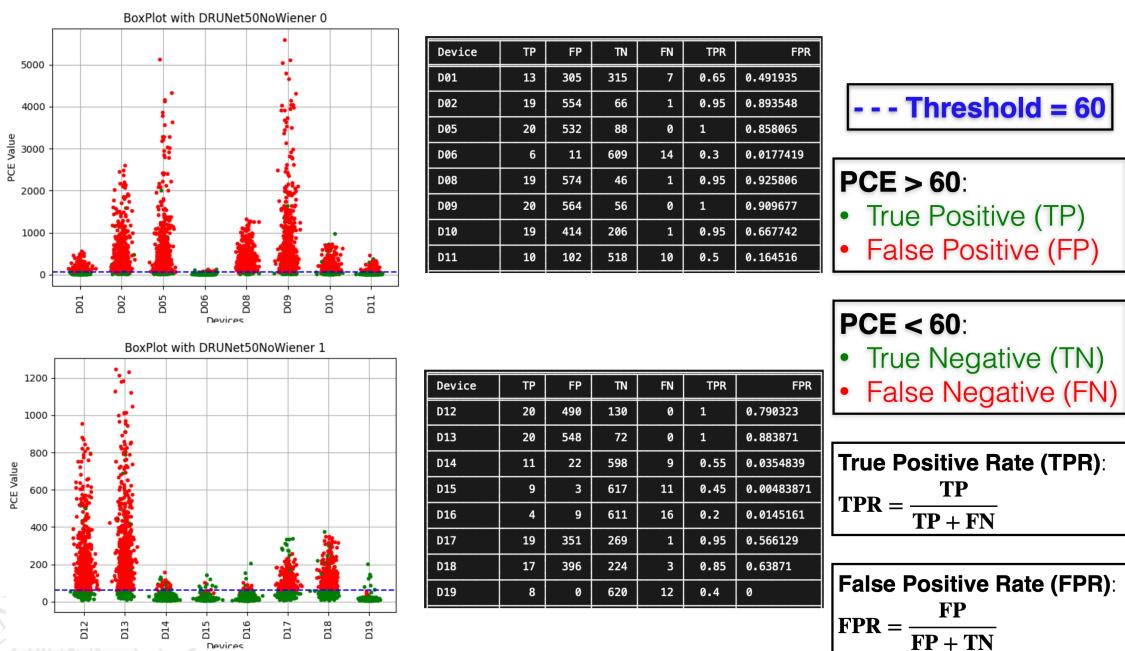


Figura 30: Parte 1/2

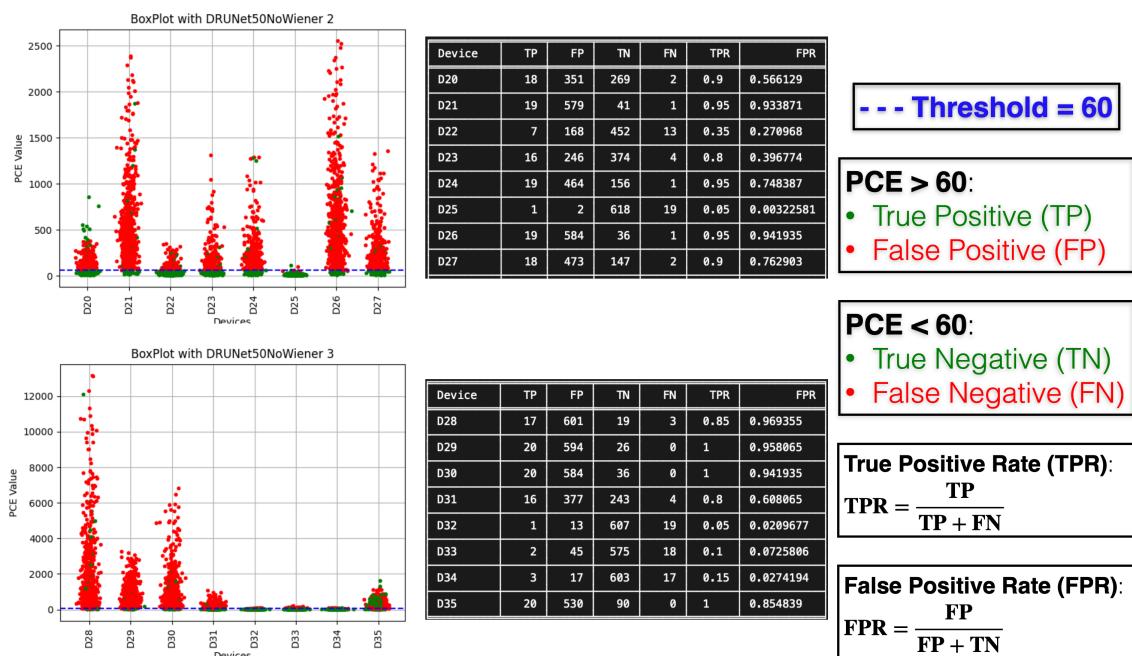


Figura 31: Parte 2/2

5.3 Rimozione di ZeroMean e di Wiener DFT

Con il termine **DRUNet50NoZeroMeanNoWiener** si intende il metodo DRUNet50 a cui sono state rimosse le operazioni di *zero_mean()* e di *wiener_dft()*. Il valore di **AUC sulla PCE** in questo caso è **0.59**. La curva ROC ottenuta è la seguente:

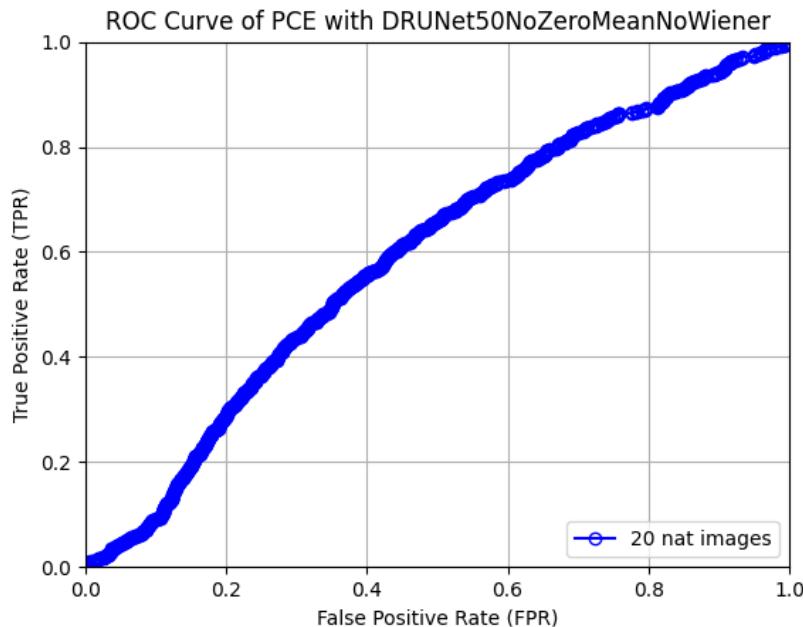


Figura 32: Curva ROC della PCE con il metodo DRUNet50NoZeroMeanNoWiener

Sono stati inoltre prodotti i seguenti grafici e tabelle riassuntive:

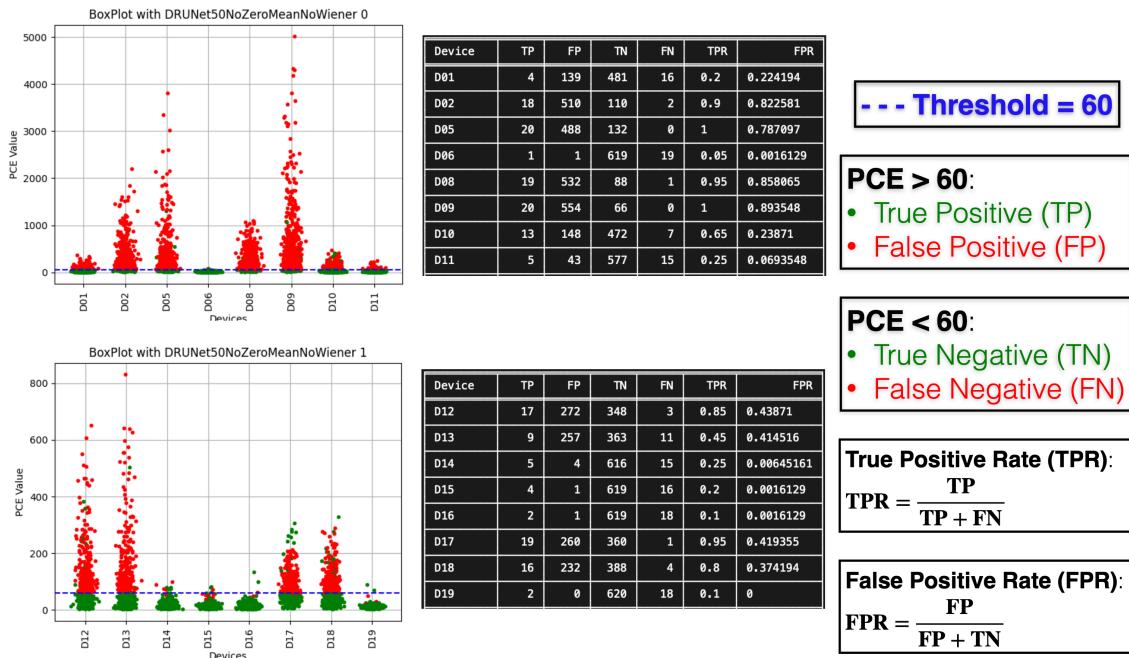


Figura 33: Parte 1/2

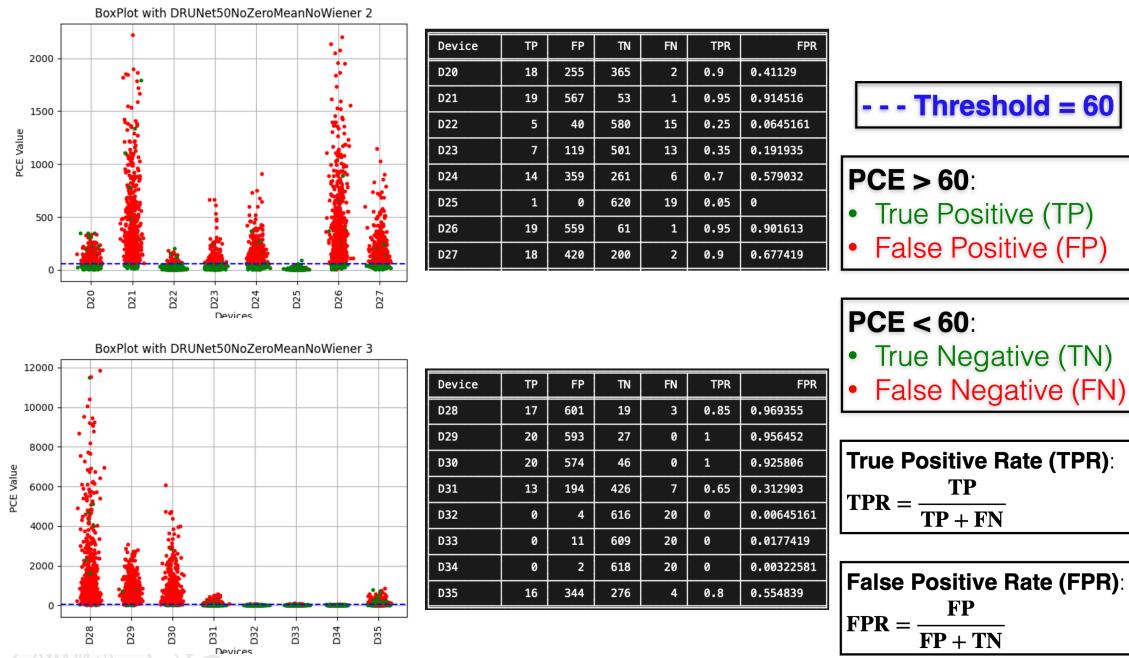


Figura 34: Parte 2/2

5.4 Confronto Risultati dell'Ablation Studies

Viene riportato adesso il confronto in termini di AUC sulla PCE tra i metodi dell'*ablation studies* sopra analizzati: **DRUNet50NoZeroMean**, **DRUNet50NoWiener** e **DRUNet50NoZeroMeanNoWiener**. Per semplicità sono riassunti nella seguente tabella:

	DRUNet50	DRUNet50NoZeroMean	DRUNet50NoWiener	DRUNet50NoZeroMeanNoWiener
AUC	0.83	0.83	0.62	0.59

Tabella 3: Confronto risultati degli *ablation studies*

Per confrontare l'andamento delle curve ROC ottenute dagli *ablation studies* con la ROC di DRUNet50, si ha di seguito il seguente grafico comparativo:

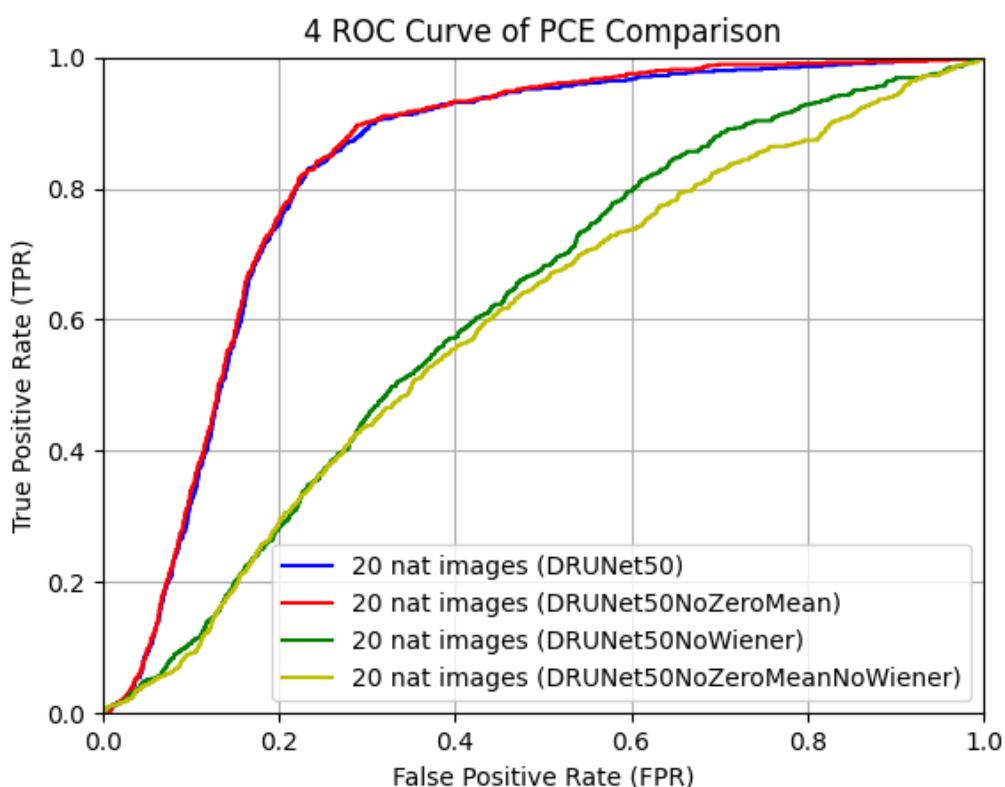


Figura 35: Confronto delle 4 curve ROC

Si può notare come la rimozione dell'**operazione di Zero Mean** non incida sulle prestazioni del sistema complessivo. La rimozione dell'**operazione di Wiener DFT** comporta invece una drastica riduzione dell'**AUC** sulla PCE.

Per una maggiore comprensione delle differenze di prestazione tra i risultati ottenuti dagli *ablation studies* e i risultati di DRUNet50, si hanno le seguenti tabelle comparative:

DRUNet50

Device	TP	FP	TN	FN	TPR	FPR
D01	20	206	414	0	1	0.33
D02	20	404	216	0	1	0.65
D05	20	499	121	0	1	0.8
D06	19	80	540	1	0.95	0.12
D08	20	428	192	0	1	0.69
D09	20	515	105	0	1	0.83
D10	20	164	456	0	1	0.26
D11	20	307	313	0	1	0.49

DRUNet50NoZeroMean

Device	TP	FP	TN	FN	TPR	FPR
D01	20	214	406	0	1	0.345161
D02	20	420	200	0	1	0.677419
D05	20	504	116	0	1	0.812903
D06	19	79	541	1	0.95	0.127419
D08	20	433	187	0	1	0.698387
D09	20	519	101	0	1	0.837697
D10	20	174	446	0	1	0.280645
D11	20	307	313	0	1	0.495161

DRUNet50NoWiener

Device	TP	FP	TN	FN	TPR	FPR
D01	13	305	315	7	0.65	0.491935
D02	19	554	66	1	0.95	0.893548
D05	20	532	88	0	1	0.858065
D06	6	11	609	14	0.3	0.0177419
D08	19	574	46	1	0.95	0.925806
D09	20	564	56	0	1	0.909677
D10	19	414	206	1	0.95	0.667742
D11	10	102	518	10	0.5	0.164516

DRUNet50NoZeroMeanNoWiener

Device	TP	FP	TN	FN	TPR	FPR
D01	4	139	481	16	0.2	0.224194
D02	18	510	110	2	0.9	0.822581
D05	20	488	132	0	1	0.787097
D06	1	1	619	19	0.05	0.0016129
D08	19	532	88	1	0.95	0.858065
D09	20	554	66	0	1	0.893548
D10	13	148	472	7	0.65	0.23871
D11	5	43	577	15	0.25	0.0693548

Figura 36: Confronto 1/4

DRUNet50

Device	TP	FP	TN	FN	TPR	FPR
D12	20	207	413	0	1	0.33
D13	20	273	347	0	1	0.44
D14	17	107	513	3	0.85	0.17
D15	20	124	496	0	1	0.2
D16	9	67	533	11	0.45	0.1
D17	19	23	597	1	0.95	0.03
D18	19	142	478	1	0.95	0.22
D19	20	99	521	0	1	0.15

DRUNet50NoZeroMean

Device	TP	FP	TN	FN	TPR	FPR
D12	20	222	398	0	1	0.358065
D13	20	308	312	0	1	0.496774
D14	17	107	513	3	0.85	0.172581
D15	20	117	503	0	1	0.18871
D16	9	68	552	11	0.45	0.109677
D17	19	25	595	1	0.95	0.0403226
D18	19	145	475	1	0.95	0.233871
D19	20	91	529	0	1	0.146774

DRUNet50NoWiener

Device	TP	FP	TN	FN	TPR	FPR
D12	20	490	130	0	1	0.790323
D13	20	548	72	0	1	0.883871
D14	11	22	598	9	0.55	0.0354839
D15	9	3	617	11	0.45	0.00483871
D16	4	9	611	16	0.2	0.0145161
D17	19	351	269	1	0.95	0.566129
D18	17	396	224	3	0.85	0.63871
D19	8	0	620	12	0.4	0

DRUNet50NoZeroMeanNoWiener

Device	TP	FP	TN	FN	TPR	FPR
D12	17	272	348	3	0.85	0.43871
D13	9	257	363	11	0.45	0.414516
D14	5	4	616	15	0.25	0.00645161
D15	4	1	619	16	0.2	0.0016129
D16	2	1	619	18	0.1	0.0016129
D17	19	260	360	1	0.95	0.419355
D18	16	232	388	4	0.8	0.374194
D19	2	0	620	18	0.1	0

Figura 37: Confronto 2/4

DRUNet50

Device	TP	FP	TN	FN	TPR	FPR
D20	18	70	550	2	0.9	0.11
D21	20	514	106	0	1	0.82
D22	20	142	478	1	0.95	0.22
D23	20	420	200	0	1	0.67
D24	20	392	228	0	1	0.63
D25	16	109	511	4	0.8	0.17
D26	20	522	98	0	1	0.84
D27	20	361	259	0	1	0.58

DRUNet50NoZeroMean

Device	TP	FP	TN	FN	TPR	FPR
D20	18	71	549	2	0.9	0.114516
D21	20	526	94	0	1	0.848387
D22	20	141	479	0	1	0.227419
D23	20	411	209	0	1	0.662903
D24	20	413	207	0	1	0.666129
D25	16	103	517	4	0.8	0.166129
D26	20	531	89	0	1	0.856452
D27	20	376	244	0	1	0.606452

DRUNet50NoWiener

Device	TP	FP	TN	FN	TPR	FPR
D20	18	351	269	2	0.9	0.566129
D21	19	579	41	1	0.95	0.933871
D22	7	168	452	13	0.35	0.270968
D23	16	246	374	4	0.8	0.396774
D24	19	464	156	1	0.95	0.748387
D25	1	2	618	19	0.05	0.00322581
D26	19	584	36	1	0.95	0.941935
D27	18	473	147	2	0.9	0.762903

DRUNet50NoZeroMeanNoWiener

Device	TP	FP	TN	FN	TPR	FPR
D20	18	255	365	2	0.9	0.41129
D21	19	567	53	1	0.95	0.914516
D22	5	40	588	15	0.25	0.0645161
D23	7	119	501	13	0.35	0.191935
D24	14	359	261	6	0.7	0.579032
D25	1	0	628	19	0.05	0
D26	19	559	61	1	0.95	0.901613
D27	18	420	200	2	0.9	0.677419

Figura 38: Confronto 3/4

DRUNet50

Device	TP	FP	TN	FN	TPR	FPR
D28	17	572	48	3	0.85	0.92
D29	20	536	84	0	1	0.86
D30	20	530	90	0	1	0.85
D31	18	309	311	1	0.95	0.49
D32	14	79	541	6	0.7	0.12
D33	14	164	456	6	0.7	0.26
D34	20	71	549	0	1	0.11
D35	20	200	420	0	1	0.32

DRUNet50NoZeroMean

Device	TP	FP	TN	FN	TPR	FPR
D28	16	572	48	4	0.8	0.922581
D29	20	539	81	0	1	0.869355
D30	20	528	92	0	1	0.851613
D31	20	315	305	0	1	0.508065
D32	14	83	537	6	0.7	0.133871
D33	15	171	449	5	0.75	0.275806
D34	20	66	554	0	1	0.106452
D35	20	210	410	0	1	0.33871

DRUNet50NoWiener

Device	TP	FP	TN	FN	TPR	FPR
D28	17	601	19	3	0.85	0.969355
D29	20	594	26	0	1	0.958065
D30	20	584	36	0	1	0.941935
D31	16	377	243	4	0.8	0.608065
D32	1	13	607	19	0.05	0.0209677
D33	2	45	575	18	0.1	0.0725806
D34	3	17	603	17	0.15	0.0274194
D35	20	530	90	0	1	0.854839

DRUNet50NoZeroMeanNoWiener

Device	TP	FP	TN	FN	TPR	FPR
D28	17	601	19	3	0.85	0.969355
D29	20	593	27	0	1	0.956452
D30	20	574	46	0	1	0.925806
D31	13	194	426	7	0.65	0.312903
D32	0	4	616	20	0	0.00645161
D33	0	11	609	20	0	0.0177419
D34	0	2	618	20	0	0.00322581
D35	16	344	276	4	0.8	0.554839

Figura 39: Confronto 4/4

6 Conclusioni

In questo paper, è stato adattato un metodo *learning-based* come **DRUNet** all'algoritmo sviluppato in Python dell'estrazione del PRNU di alcuni studenti del Politecnico di Milano, che utilizzavano invece come funzione di denoising un algoritmo *model-based*.

Questo ha portato a dei buoni risultati in termini di prestazioni di **AUC** (Area Under Curve) utilizzando come misura di correlazione la PCE (Peak to Correlation Energy).

Con il metodo *learning-based* **DRUNet100** infatti è stato possibile arrivare a un valore di **AUC pari a 0.90**, avvicinandosi molto al valore ottenuto invece con algoritmi di denoising *model-based* come **NoiseExtract**.

È stato possibile osservare che le prestazioni siano uniformi su tutti i dispositivi del dataset proposto, ad eccezione dei *device* **Huawei** che si sono rivelati i peggiori in termini di *Device Identification*, ottenendo valori di AUC sulla PCE nettamente inferiori in confronto agli altri brand.

Tramite gli **Ablation Studies**, svolti su **DRUNet50**, è stato possibile evidenziare come la rimozione dell'operazione di **Zero Mean** non influi sulle prestazioni generali del sistema complessivo. Al contrario invece la rimozione dell'operazione di **Wiener DFT** comporta un drastico calo della AUC sulla PCE.

Bibliografia

- [1] Kai Zhang - Yawei Li - Wangmeng Zuo - Lei Zhang - Luc Van Gool - Radu Timofte. *Plug-and-Play Image Restoration with Deep Denoiser Prior*. Agosto 2020.
- [2] Saeed Anwar - Nick Barnes. *Real Image Denoising with Feature Attention*. Marzo 2020.
- [3] Luca Bondi - Paolo Bestagini - Nicolò Bonettini. *Python porting of PRNU extractor and helpers functions*. Marzo 2020.