FACHHOCHSCHULE
ERFURT UNIVERSITY
OF APPLIED SCIENCES
Angewandte
Informatik

Klassische Sprachverbesserung: Wiener-Filterung, Spectral Subtraction, MMSE – Qualität vs. Verständlichkeit

Katja Fischer

Abgabedatum: 22.02.2026

Prof. Dr. Tamas Harczos

Inhaltsverzeichnis

Abbildungsverzeichnis

1 Einleitung

Verrauschte Sprachsignale sind in vielen Anwendungen ein zentrales Problem: In Hörhilfen und Cochlea-Implantaten beeinflusst Hintergrundlärm direkt die Verständlichkeit, in Telekommunikationssystemen und Videokonferenzen sinkt die wahrgenommene Qualität, und auch automatische Spracherkennungssysteme reagieren empfindlich auf Störgeräusche. Ziel der Sprachverbesserung (Speech Enhancement) ist es daher, Störanteile zu reduzieren, ohne dabei die Sprachinformation zu stark zu verzerren. Dabei stehen zwei Anforderungen häufig im Spannungsfeld: Qualität und Sprachverständlichkeit. Neben datengetriebenen, modernen Deep-Learning-Verfahren haben klassische, spektral basierte Methoden weiterhin eine hohe praktische Relevanz. Sie sind vergleichsweise leicht implementierbar, benötigen keine Trainingsdaten und sind gut interpretierbar. Insbesondere in ressourcenbeschränkten Systemen sind sie aufgrund ihres geringen Rechenaufwands weiterhin attraktiv.

In dieser Arbeit werden drei klassische Verfahren untersucht und miteinander verglichen. Zur objektiven Bewertung werden die Metriken STOI (Verständlichkeit) und PESQ (wahrgenommene Qualität) verwendet. Der Schwerpunkt liegt auf der Analyse des Trade-offs zwischen Qualität und Verständlichkeit sowie auf der Frage, welche Faktoren die erreichbare Performance begrenzen.

2 Grundlagen

Die Verbesserung verrauschter Sprachsignale stellt ein fundamentales Problem in der Audioverarbeitung dar. Das Ziel dieser Verbesserung ist die Reduktion von Störgeräuschen bei gleichzeitiger Erhaltung oder Verbesserung der Sprachqualität und -verständlichkeit. Anwendungsgebiete reichen von Hörhilfen und Telekommunikationssystemen über Sprachassistenten bis hin zu Spracherkennungssystemen.[1]

Um diese Anforderungen technisch umzusetzen, haben sich verschiedene mathematische Ansätze etabliert. Im Folgenden werden die Verfahren der spektralen Subtraktion, der Wiener-Filterung sowie der MMSE-Schätzung (Minimum Mean Square Error) betrachtet.

2.1 Grundproblem

Das grundlegende Modell der Sprachverbesserung geht davon aus, dass sich das beobachtete Signal additiv aus einem Sprach- und einem Rauschanteil zusammensetzt. [2]

$$y(n) = x(n) + d(n) \tag{2.1}$$

wobei:

- $y(n)$ das beobachtete, verrauschte Signal darstellt,
- $x(n)$ das saubere Sprachsignal ist,
- $d(n)$ das Störgeräusch repräsentiert.

Da Sprache nichtstationär ist, erfolgt die Verarbeitung typischerweise frameweise im Zeit-Frequenz-Bereich. Hierzu wird das Signal mittels Kurzzeit-Fourier-Transformation (STFT) in überlappende Zeitfenster zerlegt, im Frequenzbereich verarbeitet und anschließend durch die inverse STFT wieder in den Zeitbereich rekonstruiert.[3] Im STFT-Bereich gilt für den Zeitrahmenindex m und den Frequenzindex k: [2]

$$Y(m,k) = X(m,k) + D(m,k) \tag{2.2}$$

[1] Philipos C. Loizou, *SPEECH ENHANCEMENT Theory and Practice*, S. 29.
[2] Philipos C. Loizou, *SPEECH ENHANCEMENT Theory and Practice*, S. 93
[3] Matthias Leimeister,Csanád Egervári, Felix Kuhnke, Anja Chilian, Charlott Voigt, Tamas Harczos, *Simple spectral subtraction method enhances speech intelligibility in noise for cochlear implant listeners*.

Eine zentrale Kenngröße zur Beschreibung der Rauschbelastung ist das Signal-to-Noise Ratio (SNR). Es quantifiziert das Verhältnis der mittleren Signalleistung zur mittleren Rauschleistung und wird üblicherweise in Dezibel angegeben als

$$\mathrm{SNR}_{\mathrm{dB}} = 10 \log_{10} \left(\frac{P_{\text{Signal}}}{P_{\text{Rauschen}}} \right). \tag{2.3}$$

wobei P_{Signal} die mittlere Signalleistung und P_{Rauschen} die mittlere Rauschleistung bezeichnet. Hohe SNR-Werte entsprechen geringerer Rauschbeeinflussung, niedrige SNR-Werte einer stärkeren Störung.[4]

2.2 Spektrale Subtraktion

Der spektrale Subtraktionsalgorithmus ist eines der ersten Algorithmen für Rauschunterdrückung in Sprachsignalen.[5] Die Grundidee wurde von Boll im Jahr 1979 als Methode zur Rauschunterdrückung vorgeschlagen.[6] Der hier betrachtete Algorithmus folgt der Erweiterung nach Berouti, die das ursprüngliche Konzept um einen Oversubtraction-Faktor α und einen Spectral Floor β erweitert, um das Auftreten von *musical noise*-Artefakten zu reduzieren. Musical noise bezeichnet tonale, kurzzeitige Artefakte, die insbesondere durch binweise Verarbeitung im STFT-Bereich bei fehlerbehafteter Rauschschätzung und zu aggressiver Dämpfung entstehen.[7]

Berechnung

Die grundlegende Idee der spektralen Subtraktion besteht darin, eine Schätzung des Rausch-Leistungsspektrums $\widehat{P}_D(m,k)$ vom frameweisen Leistungsspektrum des verrauschten Signals abzuziehen: [7]

$$\widehat{P}_X(m,k) = P_Y(m,k) - \widehat{P}_D(m,k) \tag{2.4}$$

wobei $P_Y(m,k) = |Y(m,k)|^2$ die Leistungsschätzung des verrauschten Signals pro STFT-Frame m und Frequenzbin k ist.

Erweiterung nach Berouti et al.

Die einfache Subtraktion nach Gleichung (2.4) führt zu zwei Problemen: negative Werte und dem Auftreten von musical noise-Artefakten. [7] Die negativen Werte werden typischerweise durch eine Untergrenze behandelt:

$$\widehat{P}_X(m,k) = \max\big(P_Y(m,k) - \widehat{P}_D(m,k), 0\big). \tag{2.5}$$

4 Philipos C. Loizou, *SPEECH ENHANCEMENT Theory and Practice*, S. 479–482.
5 Philipos C. Loizou, *SPEECH ENHANCEMENT Theory and Practice*, S. 93.
6 Steven F. Boll, Member, „Suppression of Acoustic Noise in Speech Using Spectral Subtraction“.
7 M. Berouti, R. Schwartz, and J. Makhoul, „ENHANCEMENT OF SPEECH CORRUPTED BY ACOUSTIC NOISE“, S. 93

Zur Reduktion von *musical noise* führten Berouti et al. zwei wesentliche Erweiterungen ein. Den Oversubtraction-Faktor $\alpha \geq 1$ und den Spectral Floor β mit $0 < \beta \ll 1$: [8]

$$P_{\text{sub}}(m,k) = P_Y(m,k) - \alpha \cdot \widehat{P}_D(m,k), \tag{2.6}$$

$$\widehat{P}_X(m,k) = \max\left(P_{\text{sub}}(m,k), \beta \cdot \widehat{P}_D(m,k)\right). \tag{2.7}$$

Der Over-Subtraction-Faktor α skaliert die vom Sprachspektrum abzuziehende Rauschschätzung und bestimmt damit die Stärke der spektralen Subtraktion. Größere Werte von α>1 senken die verbleibenden Rauschpeaks stärker und können sowohl breitbandiges Rauschen als auch musical-noise-Artefakte reduzieren. Wenn α jedoch zu groß ist, kann die dadurch entstehende spektrale Verzerrung zunehmen und die Sprachverständlichkeit beeinträchtigen. Der Spectral-Floor-Parameter β setzt eine Untergrenze für das bereinigte Spektrum. Dadurch werden sehr niedrige spektrale Bereiche nicht auf Null abgesenkt, sondern angehoben. Das reduziert starke Kontraste im Restspektrum, sodass schmale verbleibende spektrale Spitzen weniger auffallen und *musical noise* durch Maskierung abgeschwächt wird. [8]

Eigenschaften

Ein wesentlicher Vorteil der spektralen Subtraktion ist ihre geringe Rechenkomplexität. Dadurch eignet sie sich besonders für Echtzeit- und eingebettete Systeme mit begrenzten Ressourcen, beispielsweise Hörgeräte und andere mobile Hörsysteme.[9]

2.3 Wiener Filterung

Die Wiener-Filterung zählt zu den klassischen Verfahren der statistischen Sprachsignalverbesserung. Im Kern basiert sie auf der Minimierung des mittleren quadratischen Fehlers (Minimum Mean-Square Error - MMSE) und lässt sich als statistisch optimaler linearer Filter interpretieren.[10] Im Gegensatz zur spektralen Subtraktion, bei der das geschätzte Rauschspektrum direkt subtrahiert wird, erfolgt hier eine frequenzabhängige Dämpfung des verrauschten Spektrums. Dies geschieht über eine Verstärkungsfunktion (Gain), die das Verhältnis von Sprach- zu Rauschleistung dynamisch berücksichtigt.[11]

[8] M. Berouti, R. Schwartz, and J. Makhoul, „ENHANCEMENT OF SPEECH CORRUPTED BY ACOUSTIC NOISE“, S. 93

[9] Iko Pieper, Amelie J. Hintermaier, Tamas Harczos, „NOISE REDUCTION FOR AUDIO IN REALTIME AND WITH LOW POWER CONSUMPTION“.

[10] Philipos C. Loizou, *SPEECH ENHANCEMENT Theory and Practice*, S. 203.

[11] Philipos C. Loizou, *SPEECH ENHANCEMENT Theory and Practice*, S. 137.

Berechnung

In ihrer Standardform ergibt sich die Übertragungsfunktion des Wiener-Filters (G) aus dem Verhältnis der Leistungsspektren von Sprachsignal P_X und Rauschsignal P_D [12,:13]

$$G(m,k) = \frac{P_X(m,k)}{P_X(m,k) + P_D(m,k)}. \tag{2.8}$$

Durch die Einführung des a-priori SNR $\xi(m,k) = P_X(m,k)/P_D(m,k)$ lässt sich diese Gleichung äquivalent formulieren als:

$$G(m,k) = \frac{\xi(m,k)}{1 + \xi(m,k)}. \tag{2.9}$$

Da das Sprach-Leistungsspektrum $P_X(m,k)$ in realen Szenarien unbekannt ist, muss die a-priori-SNR geschätzt werden. Ein weit verbreiteter Standard ist hierfür der von Ephraim und Malah eingeführte Decision-Directed-Ansatz:

$$\gamma(m,k) = \frac{|Y(m,k)|^2}{\widehat{P}_D(m,k)}, \tag{2.10}$$

$$\xi(m,k) = \lambda \frac{|\widehat{X}(m-1,k)|^2}{\widehat{P}_D(m,k)} + (1-\lambda)\max(\gamma(m,k) - 1, 0), \tag{2.11}$$

wobei λ (≈ 0.98) als Glättungsfaktor dient.[15]

Eigenschaften

Die Wiener-Filterung realisiert im STFT-Bereich eine multiplikative, frequenzselektive Dämpfung. Ein zentraler Einflussfaktor für die Sprachverbesserung ist die Qualität der Rauschleistungs- und SNR-Schätzung. Dabei kann es zu Verbesserungen der Qualität führen, aber auch zu Überdämpfung oder unzureichender Rauschreduktion. Der Rechenaufwand bleibt trotz der zusätzlichen SNR-Schätzung vergleichsweise gering, da nach der STFT im Wesentlichen elementweise Operationen pro Zeit-Frequenz-Bin durchgeführt werden.[14]

2.4 MMSE

Im Rahmen der MMSE-Methoden werden unterschiedliche Schätzgrößen betrachtet, wobei entweder die spektrale Amplitude oder deren logarithmische Darstellung minimiert wird.

[12] Yariv Ephraim, David Malah, Member, „Speech Enhancement Using a- Minimum Mean- Square Error Short-Time Spectral Amplitude Estimator“, S. 93

[13] Philipos C. Loizou, *SPEECH ENHANCEMENT Theory and Practice*, S. 146.

[14] Philipos C. Loizou, *SPEECH ENHANCEMENT Theory and Practice*, S. 140–156, 377–378.

2.4.1 MMSE-STA

Während die Wiener-Filterung den mittleren quadratischen Fehler des gesamten komplexen Spektrums minimiert, konzentriert sich die MMSE-STSA-Schätzung (Minimum Mean Square Error Short-Time Spectral Amplitude) gezielt auf die Schätzung der Amplitudenwerte.

Berechnungen

Der Algorithmus basiert auf der Annahme, dass die spektralen Koeffizienten von Sprache und Rauschen als statistisch unabhängige, komplexwertige Gauß-Zufallsvariablen modelliert werden können. Aus dieser Annahme folgt, dass die Amplituden der Sprachspektralkoeffizienten einer Rayleigh-Verteilung folgen. Das Ziel der MMSE-STSA-Schätzung ist es, einen Schätzer $\widehat{A}(m,k)$ zu finden, der den Erwartungswert des quadratischen Fehlers zwischen der wahren Amplitude A und der geschätzten Amplitude $\widehat{A}$ minimiert:[15]

$$\min E\left\{(A(m,k)-\widehat{A}(m,k))^2 \mid Y(m,k)\right\}. \tag{2.12}$$

Der optimale MMSE-Schätzer minimiert den bedingten mittleren quadratischen Fehler und ist daher der bedingte Erwartungswert der Amplitude:

$$\widehat{A}(m,k) = E[A(m,k) \mid Y(m,k)]\,. \tag{2.13}$$

Die Schätzung kann als Multiplikation der Beobachtungsamplitude mit einer zeit-frequenzabhängigen Verstärkungsfunktion geschrieben werden. Diese Gain-Funktion $G_{\text{MMSE}}(m,k)$ hängt vom a-priori SNR $\xi(m,k)$ und dem a-posteriori SNR $\gamma(m,k)$ ab und lässt sich in geschlossener Form über modifizierte Bessel-Funktionen $I_0(\cdot)$ und $I_1(\cdot)$ ausdrücken:

$$G_{\text{MMSE}}(m,k) = \frac{\sqrt{\pi}}{2}\,\frac{\sqrt{v(m,k)}}{\gamma(m,k)}\,\exp\!\left(-\frac{v(m,k)}{2}\right)\left[(1+v(m,k))\,I_0\!\left(\frac{v(m,k)}{2}\right)+v(m,k)\,I_1\!\left(\frac{v(m,k)}{2}\right)\right]. \tag{2.14}$$

Dabei ist die Hilfsvariable

$$v(m,k) = \frac{\xi(m,k)}{1+\xi(m,k)}\,\gamma(m,k) \tag{2.15}$$

definiert.[16]

Wie beim Wiener-Filter wird auch hier die Schätzung des a-priori SNR $\xi(m,k)$ üblicherweise über den Decision-Directed-Ansatz realisiert. [15]

Eigenschaften

Ein wesentlicher Vorteil der MMSE-STSA-Schätzung gegenüber der Wiener-Filterung und der spektralen Subtraktion ist die geringere Ausprägung von *musical noise*. Je nach Qualität der Rausch- und SNR-Schätzung sowie der Parametrisierung können jedoch weiterhin Restgeräusche hörbar bleiben. In der Praxis führt dies häufig zu einem natürlicheren Klangbild und

[15] Yariv Ephraim, David Malah, Member, „Speech Enhancement Using a- Minimum Mean- Square Error Short-Time Spectral Amplitude Estimator“

[16] Philipos C. Loizou, *SPEECH ENHANCEMENT Theory and Practice*, S. 211–214.

zu geringerer Sprachverzerrung. Dem steht ein moderat erhöhter Rechenaufwand gegenüber, da die Gain-Funktion modifizierte Bessel-Funktionen beinhaltet.[17] [18]

2.4.2 Log-MMSE

Der Log-MMSE-Ansatz baut auf dem gleichen statistischen Rahmen wie MMSE-STSA auf, verwendet jedoch ein anderes Optimierungskriterium. Statt den Fehler der linearen spektralen Amplitude zu minimieren, wird der Fehler im logarithmischen Amplitudenbereich betrachtet.[18]

Berechnungen

Formal ergibt sich auch hier eine zeit-frequenzabhängige Gain-Funktion, die, analog zu MMSE-STSA, von a-priori SNR $\xi(m,k)$ und a-posteriori SNR $\gamma(m,k)$ bzw. der Hilfsgröße $\nu(m,k)$ abhängt. Der entscheidende Unterschied ist, dass die geschlossene Form des Gains ein Exponentialintegral enthält:[19]

$$G_{\text{LSA}}(m,k) = \frac{\xi(m,k)}{1+\xi(m,k)} \exp\left(\frac{1}{2} E_1\big(\nu(m,k)\big)\right), \tag{2.16}$$

Eigenschaften

Im Vergleich zur MMSE-STSA-Schätzung führt die logarithmische Fehlergewichtung häufig zu einer stärkeren Dämpfung in spektralen Tälern und damit zu einer subjektiv weniger tonalen Restgeräuschwahrnehmung, wobei der genaue Trade-off weiterhin von der SNR-Schätzung und der Parametrisierung abhängt. Der zusätzliche Rechenaufwand entsteht primär durch die Auswertung bzw. Approximation des Exponentialintegrals.[20]

2.5 Evaluierung der Sprachverbesserung

Zur Beurteilung der Leistungsfähigkeit von Sprachverbesserungsalgorithmen werden in dieser Arbeit objektive Metriken eingesetzt. Diese berechnen die Verzerrungen zwischen Referenz- und Ausgabesignal und schließen daraus auf wahrgenommene Qualität bzw. Verständlichkeit. Dabei wird STOI für Verständlichkeit und PESQ für wahrgenommene Qualität verwendet. [21,22]

[17] Yariv Ephraim, David Malah, Member, „Speech Enhancement Using a- Minimum Mean- Square Error Short-Time Spectral Amplitude Estimator“.
[18] Philipos C. Loizou, *SPEECH ENHANCEMENT Theory and Practice*, S. 225–227
[19] Philipos C. Loizou, *SPEECH ENHANCEMENT Theory and Practice*, S. 228–229.
[20] Philipos C. Loizou, *SPEECH ENHANCEMENT Theory and Practice*, S. 229–231.
[21] Philipos C. Loizou, *SPEECH ENHANCEMENT Theory and Practice*, S. 505–506
[22] Yi Hu and Philipos C. Loizou, Senior Member, IEEE, „Evaluation of Objective Quality Measuresfor Speech Enhancement“.

STOI (Short-Time Objective Intelligibility)

Die Short-Time Objective Intelligibility (STOI) ist eine intrusive objektive Metrik zur Vorhersage der Sprachverständlichkeit. Sie benötigt sowohl das saubere Referenzsignal als auch das verarbeitete Signal. STOI liefert einen skalaren Wert im Bereich $[0, 1]$, wobei höhere Werte eine höhere erwartete Verständlichkeit anzeigen.[23]

PESQ (Perceptual Evaluation of Speech Quality)

Zur Bewertung der wahrgenommenen Sprachqualität wird Perceptual Evaluation of Speech Quality (PESQ) verwendet. PESQ ist eine standardisierte intrusive Metrik, die auf einem psychoakustischen Modell basiert und Verzerrungen sowie wahrgenommene Artefakte im Signal berücksichtigt. Die Skala reicht von -0.5 bis 4.5, wobei höhere Werte eine bessere wahrgenommene Qualität anzeigen.[24]

Da Sprachverbesserungsalgorithmen Qualität und Verständlichkeit unterschiedlich beeinflussen können, ermöglicht die gemeinsame Betrachtung beider Metriken eine differenzierte Bewertung der Ergebnisse.

[23] Cees H. Taal, Richard C. Hendriks, Richard Heusdens, and Jesper Jensen, „An Algorithm for Intelligibility Prediction ofTime–Frequency Weighted Noisy Speech“.

[24] J. G. Beerends , R. A. van Buuren , J. M. Van Vugt , J.A. Verhave, „PESQ Based Speech Intelligibility Measurement“.

3 Umsetzung und Implementierung

Die Implementierung der Rauschunterdrückungsalgorithmen erfolgte in Python 3.13[1] unter Verwendung von Bibliotheken für Audioverarbeitung und numerische Berechnungen. Das System wurde als modulares Framework konzipiert, das verschiedene Algorithmen über eine einheitliche Schnittstelle integriert.

3.1 Code-Struktur und Abhängigkeiten

Das System besteht aus mehreren miteinander verbundenen Modulen:

- `speech_enhancement_comparison.py`: Zentrale Pipeline zur Parameteroptimierung, Batch-Auswertung, Speicherung optimierter Audiodateien und Ausgabe der Ergebnisse
- `spectral_subtractor.py`: Implementierung der spektralen Subtraktion mit Oversubtraction und spektralem Floor
- `wiener_filter.py`: Wiener-Filter-Implementierung im STFT-Bereich mit Decision-Directed Schätzung des a-priori SNR
- `advanced_mmse.py`: Log-MMSE/LSA-MMSE-Implementierung mit Exponentialintegral und zusätzlicher Sprachpräsenzwahrscheinlichkeit
- `noise_estimation.py`: Rauschschätzverfahren Percentile, Min-Tracking, Oracle/TrueNoise
- `evaluation_metrics.py`: Berechnung von STOI, PESQ, SNR sowie einer kombinierten Bewertungsmetrik
- `parameter_ranges.py`: Definition der Parameterbereiche für die Optimierung je Algorithmus

Die folgenden Bibliotheken wurden zur Realisierung verwendet. Für numerische Operationen wurde NumPy 2.3.4[2] eingesetzt. Die STFT sowie ISTFT wurden mit Librosa 0.11.0[3] realisiert. Für spezielle mathematische Funktionen kam SciPy 1.16.2[4] zum Einsatz. Zur objektiven Evaluierung der Sprachverständlichkeit und -qualität wurden PySTOI 0.4.1[5] sowie das PESQ-Paket 0.0.4[6] verwendet. Die statistische Auswertung erfolgte mit Pandas 3.0.0,[7] während Matplotlib 3.10.8[8] zur Visualisierung der Ergebnisse eingesetzt wurde.

1 *Python 3.13.0*.
2 NumPy Developers, *NumPy 2.3.4*.
3 Brian McFee, librosa development team, *librosa 0.11.0*.
4 SciPy Developers, *scipy 1.16.2*.
5 Manuel Pariente, *pystoi*.
6 ludlows, *pesq 0.0.4*.
7 The Pandas Development Team, *pandas 3.0.0*.
8 John D. Hunter, Michael Droettboom, *matplotlib 3.10.8*.

3.2 Implementierte Algorithmen

Zur praktischen Untersuchung der in Kapitel 2 beschriebenen Verfahren wurden drei Algorithmen implementiert.

Erweiterte Spektrale Subtraktion

Die Implementierung folgt der Erweiterung nach Berouti.[9]

Wiener-Filter

Der implementierte Wiener-Filter entspricht dem klassischen STFT-basierten Gain-Ansatz. Die erforderliche a-priori-SNR $\xi(m, k)$ wird rekursiv mittels Decision-Directed-Schätzung bestimmt, wodurch eine zeitliche Glättung der SNR-Schätzung erreicht wird.[10] Zur Stabilisierung wird eine Untergrenze für den Gain verwendet (gain_floor).

Log-MMSE

Als MMSE-Variante wurde der Log-MMSE implementiert. In der Implementierung wird zusätzlich eine Sprachpräsenzwahrscheinlichkeit berücksichtigt, um die Gain-Anwendung in sprachlosen Zeit-Frequenz-Bereichen adaptiv zu steuern.

3.3 Parameteroptimierung

Für jeden Algorithmus wird eine systematische Parameteroptimierung durchgeführt, die drei Zielfunktionen maximiert:

- **STOI-Optimierung**: Maximierung der Sprachverständlichkeit
- **PESQ-Optimierung**: Maximierung der wahrgenommenen Sprachqualität
- **Balance-Optimierung**: Maximierung einer kombinierten Metrik

$$\text{Score}_{\text{comb}} = 0.5 \cdot \text{STOI} + 0.5 \cdot \frac{\max(0, \text{PESQ})}{4.5}$$

Die Optimierung erfolgt als Grid Search in einem festgelegten Parameterraum. Die algorithmusspezifischen Parameterbereiche sind in `parameter_ranges.py` definiert.

Optimiert werden sowohl gemeinsame als auch algorithmusspezifische Parameter. Gemeinsam für alle Verfahren sind die STFT-Parameter (`n_fft`, `hop_length`) sowie Parameter der Rauschschätzung (`noise_percentile`, `noise_method`). Darüber hinaus werden je Algorithmus folgende Parameter optimiert:

[9] M. Berouti, R. Schwartz, and J. Makhoul, „ENHANCEMENT OF SPEECH CORRUPTED BY ACOUSTIC NOISE“.
[10] Yariv Ephraim, David Malah, Member, „Speech Enhancement Using a- Minimum Mean- Square Error Short-Time Spectral Amplitude Estimator“.

- **Spektrale Subtraktion**: α (Oversubtraction), β (spektraler Floor)
- **Wiener-Filter**: α (Decision-Directed Glättung), `gain_floor`
- **OM-LSA**: α (Decision-Directed Glättung), $\xi_{\min}$, q (a-priori Sprachabwesenheitswahrscheinlichkeit), μ (Rausch-Glättung), `gain_floor`

Die Kernfunktion `optimize_parameters` implementiert den Optimierungsablauf:

1. Berechnung der Baseline-Metriken (STOI, PESQ, SNR) für das unverarbeitete verrauschte Signal
2. Generierung aller Parameterkombinationen
3. Iteration über alle Kombinationen:
   a) Anwendung des Algorithmus mit den aktuellen Parametern
   b) Nachbearbeitung des Ergebnissignals: zeitliche Ausrichtung zur Referenz mittels Kreuzkorrelation (max. $\pm 0.10\,\mathrm{s}$, Korrelation über die ersten $\approx 2\,\mathrm{s}$), sowie Längenausgleich und Monokonvertierung
   c) Berechnung von STOI, PESQ, SNR sowie des kombinierten Scores
   d) Aktualisierung der besten Parameterkonfigurationen für STOI-, PESQ- und Balance-Optimierung
4. Rückgabe der optimalen Parameter und der zugehörigen verarbeiteten Audiosignale pro Zielfunktion

Zur Erhöhung der Robustheit werden Schutzmaßnahmen eingesetzt. Clipping der Audiosamples auf $[-1, 1]$, Prüfung auf numerische Stabilität, sowie Längenausgleich mit dem Referenzsignal.

3.4 Rauschschätzung

In den Ergebnisabschnitten wird zwischen *True Noise* und geschätztem Noise unterschieden. Wenn im Folgenden von *True Noise* die Rede ist, wird die Ground-Truth-Rauschinformation verwendet. Diese Auswertung dient als theoretische Obergrenze der erreichbaren Performance unter idealen Bedingungen.

Wenn hingegen kein *True Noise* verwendet wird, erfolgt die Rauschschätzung ausschließlich über *Percentile* und *Min-Tracking*. Es werden dabei pro Datensatz beide Schätzverfahren getestet und anschließend das jeweils bessere Ergebnis, gemessen an der Zielmetrik der Optimierung, für die Auswertung herangezogen.

Percentile-basierte Schätzung (statisch)

Die Percentile-Methode liefert eine zeitlich konstante Schätzung der Rauschleistungsdichte (Noise-PSD) der Form $(n_{\mathrm{bins}}, 1)$. Hierzu wird für jedes Frame eine „Quietness“-Kenngröße (Leisheitsmaß) aus der gemittelten Log-Energie berechnet, um sprach- bzw. signalärmere Abschnitte zu identifizieren. Auf Basis des gewählten Perzentils werden anschließend die k energieärmsten Frames selektiert; die Anzahl k wird dabei durch `min_frames` nach unten und durch `max_fraction` nach oben begrenzt. Die finale Noise-PSD ergibt sich, indem für jeden Frequenzbin das entsprechende Perzentil über die ausgewählten „leisen“ Frames bestimmt wird.

Min-Tracking (adaptiv)

Das Min-Tracking liefert eine zeitabhängige Schätzung der Rauschleistungsdichte (Noise-PSD) der Form $(n_{\mathrm{bins}}, n_{\mathrm{frames}})$. Ausgangspunkt ist das Power-Spektrogramm, das zur Robustheit zunächst entlang der Zeitachse mittels IIR-Glättung geglättet wird. Der Glättungsfaktor α ist über `smoothing_factor` vorgegeben. Im Anschluss wird für jeden Frequenzbin ein gleitender Minimum-Filter über ein Fenster der Länge `window_size` angewendet. Die so verfolgten lokalen Minima approximieren die rauschdominierte spektrale Leistung und werden als Noise-PSD verwendet. Zur numerischen Stabilisierung wird zusätzlich ein spektraler Floor relativ zum Medianspektrum gesetzt, um unrealistisch kleine Werte und ein Kollabieren gegen Null zu verhindern.

3.5 Testdaten

Verwendet wurde die *Noisy speech database for training speech enhancement algorithms and TTS models* der University of Edinburgh.[11] Die verrauschten Signale wurden synthetisch erzeugt, indem saubere Sprachaufnahmen mit Umgebungsgeräuschen gemischt wurden. Zu jeder Aufnahme liegt eine korrespondierende Clean-Referenz vor, sodass STOI und PESQ referenzbasiert berechnet werden können. Für die Reproduzierbarkeit wurden alle Audiosignale einheitlich vorverarbeitet: Monokonvertierung, Resampling auf $16\,\mathrm{kHz}$ sowie zeitlicher und Längenabgleich von Clean- und Noisy-Signalen.

[11] Valentini-Botinhao, Cassia, *Noisy speech database for training speech enhancement algorithms and TTS models*.

4 Ergebnisse und Diskussion

Dieses Kapitel fasst die Erkenntnisse aus der statistischen Auswertung zusammen und ordnet sie im Kontext der Implementierung und Optimierungsstrategie ein.

4.1 Gesamtvergleich der Algorithmen

Ziel dieses Abschnitts ist ein grundlegender, algorithmusübergreifender Vergleich der drei betrachteten Verfahren. Für diesen Vergleich wurden alle verfügbaren Testdaten berücksichtigt. Pro Algorithmus gingen insgesamt 100 Datensätze in die Auswertung ein. Jeder Datensatz entspricht dabei einem optimierten Ergebnis. Die Rauschschätzung erfolgte dabei ausschließlich über die implementierten Schätzverfahren Min-Tracking beziehungsweise Percentile.

STOI-Optimierung

Abbildung 4.1 zeigt die durchschnittlichen besten STOI-Werte für die drei betrachteten Algorithmen.

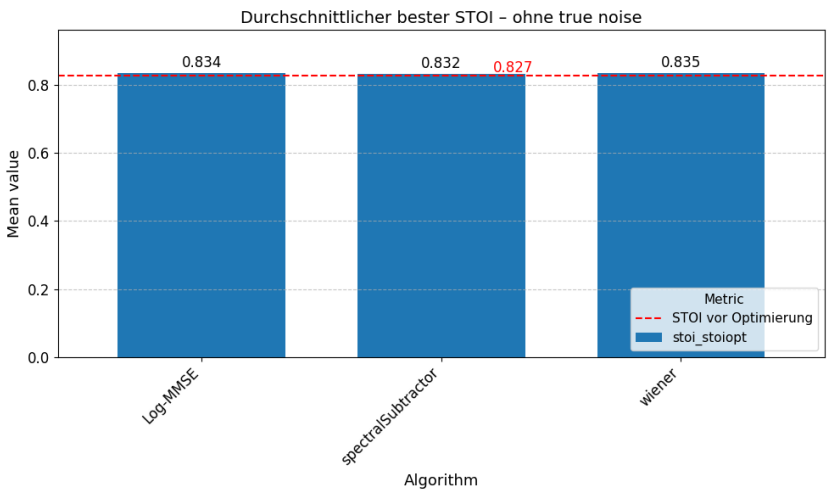

Abbildung 4.1: Durchschnittlicher bester STOI Wert pro Algorithmus

Die Verbesserungen im Vergleich zur Baseline fallen moderat aus. Es ist aber auch zu erkennen, dass die Baseline bereits vor der Optimierung auf einem hohen Niveau ist.

PESQ-Optimierung

Abbildung 4.2 zeigt die durchschnittlichen besten PESQ-Werte für die drei betrachteten Algorithmen.

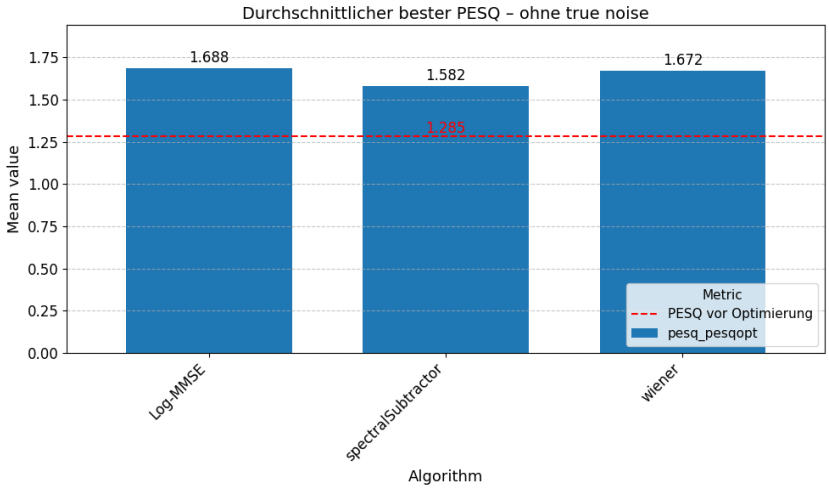

Abbildung 4.2: Durchschnittlicher bester PESQ Wert pro Algorithmus

Das Diagramm zeigt, dass alle Verfahren die wahrgenommene Sprachqualität im Mittel gegenüber dem unbearbeiteten Signal steigern. Log-MMSE erzielt hierbei den höchsten durchschnittlichen PESQ-Wert, dicht gefolgt vom Wiener-Filter. Die spektrale Subtraktion erzielte ein etwas niedrigeres Ergebnis.

4.2 Gesamtvergleich der Algorithmen mit True Noise

Im vorherigen Abschnitt erfolgte die Rauschschätzung ausschließlich über realistische Verfahren wie Min-Tracking oder Percentile. Um das theoretische Leistungspotenzial der Algorithmen zu untersuchen, wird im Folgenden ein Vergleich unter Verwendung von True Noise durchgeführt. Dabei wird das tatsächliche Rauschspektrum verwendet, wobei diese Annahme in realen Anwendungen nicht praktikabel ist. Sie stellt eine obere Leistungsgrenze dar und erlaubt eine Einordnung der zuvor erzielten Ergebnisse. Für diese Auswertung wurden insgesamt 81 Datensätze pro Algorithmus berücksichtigt.

STOI

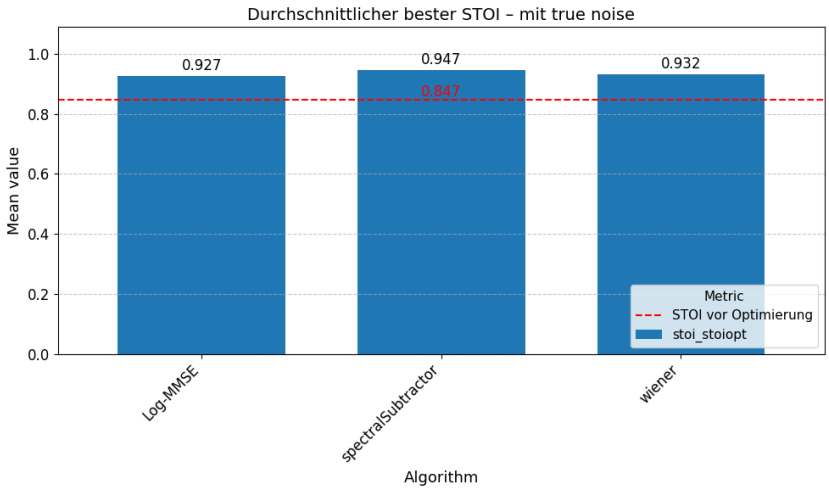

Abbildung 4.3: Durchschnittlicher bester STOI-Wert pro Algorithmus mit True Noise

Im Vergleich zur Rauschschätzung sind die Verbesserungen hier deutlich stärker ausgeprägt. Besonders der Spectral Subtractor profitiert erheblich von der exakten Kenntnis des Rauschanteils und erzielt den höchsten mittleren STOI-Wert. Dies zeigt, dass die Genauigkeit der Rauschschätzung einen erheblichen Einfluss auf die erreichbare Sprachverständlichkeit hat. Während die Algorithmen zuvor nur moderate Verbesserungen erzielten, wird unter idealen Bedingungen eine signifikante Steigerung möglich.

Um diesen Effekt genauer einzuordnen, zeigt Abbildung 4.4 den Performance-Gap zwischen True Noise und geschätztem Noise, aufgeschlüsselt nach Szenario.

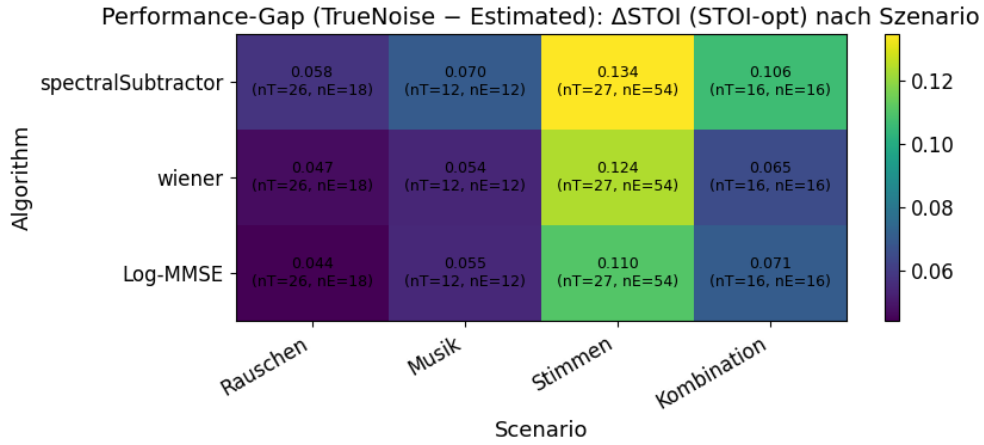

Abbildung 4.4: Performance-Gap (True Noise minus Estimated): ΔSTOI (STOI-Optimiert) nach Szenario

Auffällig ist, dass der Unterschied im Szenario *Stimmen* am größten ausfällt. Das könnte darauf hindeuten, dass die Rauschschätzung bei sprachähnlichen Störquellen weniger zuverlässig ist. Die geringsten Unterschiede zeigen sich in den Szenarien *Rauschen* und *Musik*. Das spricht dafür, dass die verwendeten Rauschschätzverfahren in diesen Fällen überwiegend zuverlässige

Noise-PSD-Schätzungen liefern und die Ergebnisse daher näher an der True Noise-Referenz liegen. Zudem wird deutlich, dass der Spectral Subtractor am stärksten von True Noise profitiert. Das deckt sich mit Abbildung 4.3.

PESQ

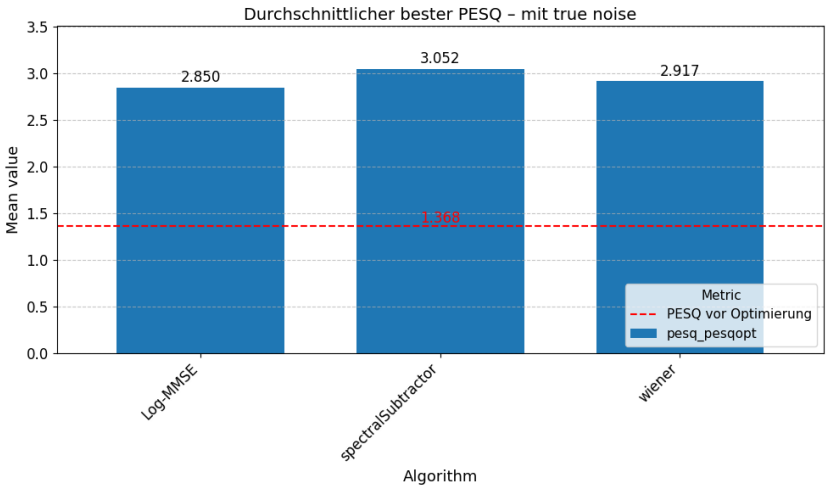

Abbildung 4.5: Durchschnittlicher bester PESQ-Wert pro Algorithmus mit True Noise

Im Gegensatz zu der vorherigen Auswertung mit geschätztem Rauschen ist hier eine massive Verbesserung der wahrgenommenen Sprachqualität erkennbar. Auch hier zeigt sich, dass insbesondere die spektrale Subtraktion stark von der exakten Kenntnis des Rauschspektrums profitiert. Der Leistungsunterschied zwischen den Algorithmen bleibt jedoch insgesamt moderat.

Der Vergleich mit True Noise verdeutlicht das theoretische Maximum der betrachteten Verfahren. Die deutlich höheren STOI- und PESQ-Werte zeigen, dass die Limitierung der Algorithmen in realistischen Szenarien maßgeblich durch die Qualität der Rauschschätzung bestimmt wird. Während sich die Algorithmen bei geschätztem Rauschen nur moderat unterscheiden, wird unter True Noise-Bedingungen sichtbar, dass alle Verfahren grundsätzlich zu deutlich höheren Qualitäts- und Verständlichkeitswerten fähig sind.

4.3 Trade-off: Verständlichkeit vs. Qualität

Neben der isolierten Optimierung einzelner Metriken stellt sich die zentrale Frage, wie sich Sprachverständlichkeit (STOI) und wahrgenommene Qualität (PESQ) gegenseitig beeinflussen. Ziel dieses Abschnitts ist es, den Zusammenhang beider Größen systematisch zu untersuchen und mögliche Zielkonflikte sichtbar zu machen. Hierzu werden die mittleren Differenzen relativ zum unbearbeiteten Signal betrachtet. Die Auswertung basiert jeweils auf 100 Datensätzen pro Algorithmus. In den folgenden Diagrammen sind sowohl die Einzelwerte pro Datei als auch die jeweiligen Mittelwerte dargestellt.

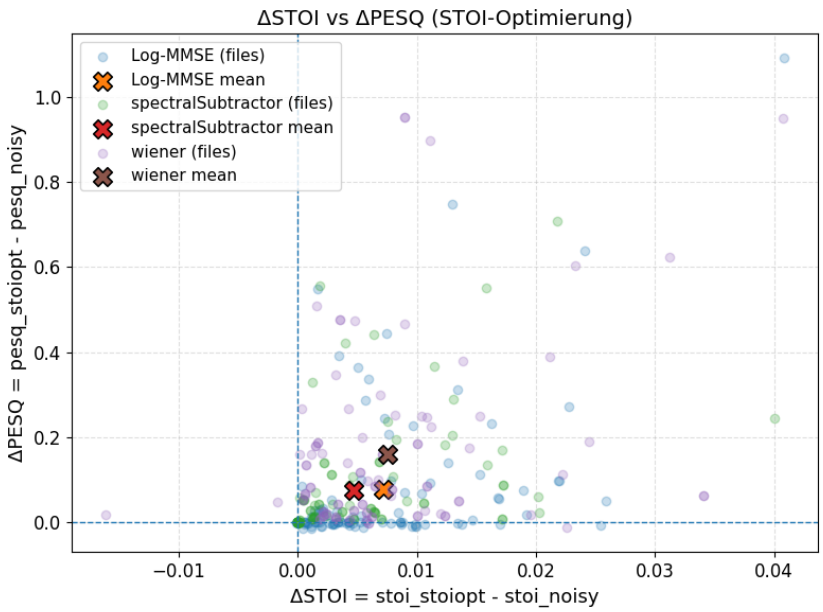

Abbildung 4.6: Zusammenhang zwischen ΔSTOI und ΔPESQ bei STOI-Optimierung

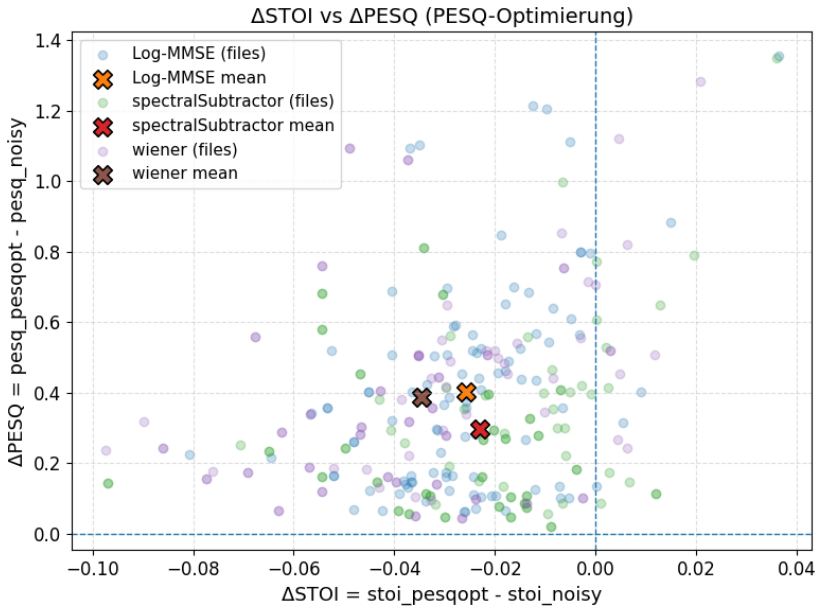

Abbildung 4.7: ΔSTOI vs. ΔPESQ bei STOI-Optimierung

Abbildung 4.6 zeigt die Veränderungen bei gezielter STOI-Optimierung, Abbildung 4.7 die entsprechenden Ergebnisse bei PESQ-Optimierung. Ein direkter Vergleich beider Darstellungen verdeutlicht den strukturellen Unterschied der Optimierungsstrategien.

Man kann erkennen, dass die Werte bei PESQ im Gegensatz zu STOI vertikal nach links auf der Achse verschoben sind. Gleichzeitig, sind die Punkte deutlich weiter nach oben verschoben. Die Qualitätssteigerung wird hier also durch eine Reduktion der objektiven Sprachverständlichkeit erreicht.

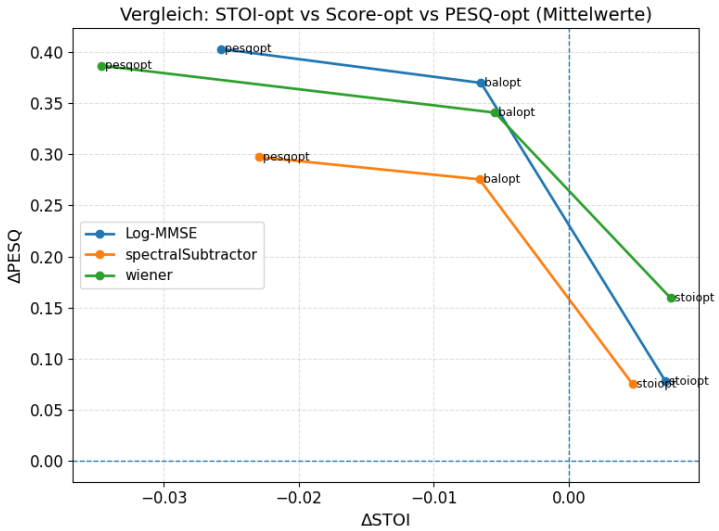

Abbildung 4.8: Vergleich der mittleren ΔSTOI- und ΔPESQ-Werte für STOI-, PESQ- und balancierte Optimierung

Abbildung 4.8 fasst die Mittelwerte der drei Optimierungsstrategien zusammen: STOI-Optimierung, PESQ-Optimierung sowie die kombinierte Optimierung. Hier wird der Zielkonflikt besonders deutlich. Während die PESQ-Optimierung die größten Qualitätsgewinne erzielt, geht dies mit negativen Veränderungen in der Verständlichkeit einher. Die STOI-Optimierung verbessert hingegen primär die Verständlichkeit bei vergleichsweise geringen Qualitätsgewinnen. Die balancierte Optimierung liegt erwartungsgemäß zwischen beiden Extremen und stellt einen Kompromiss dar.

Die Ergebnisse verdeutlichen einen klaren Zielkonflikt zwischen Sprachverständlichkeit und wahrgenommener Qualität. Der Trade-off ist nicht nur qualitativ erkennbar, sondern quantitativ klar nachweisbar. Die Analyse zeigt somit, dass eine gleichzeitige Maximierung beider Metriken nur eingeschränkt möglich ist und stets eine Priorisierung erforderlich bleibt.

5 Fazit

Ziel dieser Arbeit war es, drei klassische Verfahren der Sprachverbesserung zu implementieren und systematisch anhand der objektiven Metriken STOI und PESQ zu vergleichen. Die Ergebnisse zeigen, dass alle untersuchten Verfahren im Mittel Verbesserungen erzielen können- Es zeigt jedoch auch, dass die Verbesserungen nicht nur von den verwendeten Algorithmen abhängt, sondern besonders auch durch die Rauschschätzung begrenzt wird. Die Optimierung von STOI und PESQ machte den Zielkonflikt deutlich. Eine PESQ-Optimierung führt typischerweise zu größeren Qualitätsgewinnen, geht jedoch häufig mit Einbußen in STOI einher und umgekehrt. Die kombinierte Score-Optimierung stellt erwartungsgemäß einen praktikablen Kompromiss zwischen beiden Zielgrößen dar und eignet sich, wenn keine eindeutige Priorisierung vorgegeben ist. Für praktische Anwendungen folgt daraus, dass die Optimierungsstrategie konsequent am Zielsystem ausgerichtet werden sollte. Gleichzeitig ist für weitere Verbesserungen insbesondere bei schwierigen Störszenarien eine robuste Rauschschätzung ein entscheidender Hebel.

Literatur

Brian McFee, librosa development team. *librosa 0.11.0*. 11. März 2025. URL: https://pypi.org/project/librosa/ (besucht am 17. 02. 2026).

Cees H. Taal, Richard C. Hendriks, Richard Heusdens, and Jesper Jensen. „An Algorithm for Intelligibility Prediction ofTime–Frequency Weighted Noisy Speech“. In: Sep. 2011. DOI: 10.1109/TASL.2011.2114881.

Iko Pieper, Amelie J. Hintermaier, Tamas Harczos. „NOISE REDUCTION FOR AUDIO IN REALTIME AND WITH LOW POWER CONSUMPTION“. In.

J. G. Beerends , R. A. van Buuren , J. M. Van Vugt , J.A. Verhave. „PESQ Based Speech Intelligibility Measurement“. In: 2009.

John D. Hunter, Michael Droettboom. *matplotlib 3.10.8*. 10. Dez. 2025. URL: https://pypi.org/project/matplotlib/ (besucht am 17. 02. 2026).

ludlows. *pesq 0.0.4*. 17. Mai 2022. URL: https://pypi.org/project/pesq/ (besucht am 17. 02. 2026).

M. Berouti, R. Schwartz, and J. Makhoul. „ENHANCEMENT OF SPEECH CORRUPTED BY ACOUSTIC NOISE“. In: 1979. DOI: 10.1109/ICASSP.1979.1170788.

Manuel Pariente. *pystoi*. 29. Dez. 2023. URL: https://pypi.org/project/pystoi/ (besucht am 17. 11. 2025).

Matthias Leimeister,Csanád Egervári, Felix Kuhnke, Anja Chilian, Charlott Voigt, Tamas Harczos. *Simple spectral subtraction method enhances speech intelligibility in noise for cochlear implant listeners*. Dez. 2015.

NumPy Developers. *NumPy 2.3.4*. 15. Okt. 2025. URL: https://pypi.org/project/numpy/2.3.4/ (besucht am 17. 02. 2026).

Philipos C. Loizou. *SPEECH ENHANCEMENT Theory and Practice*. Second. CRC Press, 2013. ISBN: 978-1-4665-0422-6.

Python 3.13.0. 7. Okt. 2024. URL: https://www.python.org/downloads/release/python-3130/ (besucht am 17. 02. 2026).

SciPy Developers. *scipy 1.16.2*. 11. Sep. 2025. URL: https://pypi.org/project/scipy/1.16.2/ (besucht am 18. 02. 2026).

Steven F. Boll, Member. „Suppression of Acoustic Noise in Speech Using Spectral Subtraction“. In: Apr. 1979. DOI: 10.1109/TASSP.1979.1163209.

The Pandas Development Team. *pandas 3.0.0*. 21. Jan. 2026. URL: https://pypi.org/project/pandas/3.0.0/ (besucht am 17. 02. 2026).

Valentini-Botinhao, Cassia. *Noisy speech database for training speech enhancement algorithms and TTS models*. 21. Aug. 2017. URL: https://datashare.ed.ac.uk/handle/10283/2791 (besucht am 20. 10. 2025).

Yariv Ephraim, David Malah, Member. „Speech Enhancement Using a- Minimum Mean- Square Error Short-Time Spectral Amplitude Estimator“. In: 1984. DOI: 10.1109/TASSP.1985.1164550.

Yi Hu and Philipos C. Loizou, Senior Member, IEEE. „Evaluation of Objective Quality Measures-for Speech Enhancement“. In: Jan. 2008. DOI: 10.1109/TASL.2007.911054.