ACÚSTICA 98

ESTUDIO COMPARATIVO DE DISTINTOS ALGORITMOS SUPRESORES DE RUIDO

Josep M.SALAVEDRA, Xavier BOU.

Departament de Teoria del Senyal i Comunicacions. Universitat Politècnica de Catalunya.
c/ Jordi Girona 1-3, Campus Nord UPC , mòdul D5. 08034 Barcelona, España.
Tfno: +34.93.4017404. Fax: +34.93.4016447. E-mail: mia@gps.tsc.upc.es

SUMMARY

Some Speech Enhancement algorithms based on the iterative Wiener filtering Method due to Lim-Oppenheim [2] are evaluated. In the original Lim-Oppenheim algorithm, AR spectral estimation of speech is carried out using a second-order analysis, but our algorithms consider an AR estimation by means of cumulant analysis. This work extends some preceding papers due to the authors. Third- and fourth-order cumulant-based algorithms are compared to classical second-order one. This comparison is evaluated by considering three different environments. A detailed study shows that third-order cumulant-based algorithm significantly increases noise suppression and, therefore, convergence speed of this iterative algorithm is strongly accelerated.

1. INTRODUCCIÓN

La gran mayoría de aplicaciones de procesado de la señal de voz sufren una gran degradación en sus prestaciones cuando se consideran entornos reales ruidosos, en lugar de las condiciones ideales de laboratorio donde fueron diseñados. En la presente comunicación se propone un preprocesado 'front-end' encaminado a la mejora de la calidad de la voz por medio de un modelado paramétrico de la voz insensible al ruido, procurando que la pérdida de inteligibilidad ocasionada no sea significativa. Este desacoplo voz-ruido se pretende alcanzar mediante la consideración de las estadísticas de orden superior. Para procesos Gaussianos todos los cumulantes de orden superior a dos son idénticamente nulos y, además, los procesos no Gaussianos cuya p.d.f. sea simétrica presentan todos sus cumulantes de orden impar nulos [1]. Así, considerando que el ruido suele presentar una distribución Gaussiana o una p.d.f. simétrica y, además, la voz suele tener una caracterización no Gaussiana (especialmente en las tramas sonoras), entonces es posible obtener un modelado AR mucho más insensible al ruido. El algoritmo básico considerado se basa en un filtrado iterativo de Wiener, propuesto originariamente por Lim y Oppenheim [2], donde se estima el modelo AR de la voz original a partir de las estadísticas de segundo orden. En cambio, en el presente trabajo este modelo AR se estima a partir de los cumulantes de tercer (o cuarto) orden, obtenidos a partir de la señal ruidosa y, luego, se obtienen los coeficientes a_k que modelan la señal de voz original.

2. ALGORITMOS BASADOS EN CUMULANTES.

A continuación se evaluan las prestaciones obtenidas por algoritmos iterativos de Wiener fundamentados en Estadísticas de Orden Superior (HOS): el algoritmo AR3 basado en los cumulantes de tercer orden y el algoritmo AR4 basado en los cumulantes de cuarto orden. Además, se comparan sus prestaciones en relación al algoritmo iterativo de Wiener clásico, AR2, basado en los cumulantes de segundo orden o función autocorrelación.

El filtrado iterativo de Wiener ofrece, a cada iteración, una señal de voz realzada que contiene una menor presencia de ruido a medida que aumenta el número de iteraciones procesadas. No obstante no resulta aconsejable el procesado de un número importante de iteraciones puesto que cada iteración comporta una cierta distorsión adicional sobre la señal de voz, adquiriendo esta distorsión especial relevancia a partir de la segunda iteración procesada [4]. Para determinar el número de iteraciones por trama a procesar se debe llegar a un compromiso entre supresión de ruido y pérdida de inteligibilidad. Este compromiso depende, también, del nivel de ruido presente en la señal de voz. A continuación se evaluan tres situaciones distintas correspondientes a tres posibles entornos de trabajo: (a) entornos poco ruidosos, (b) entornos ruidosos y (c) entornos altamente ruidosos. En cada uno de estos entornos se evaluan y comparan las prestaciones ofrecidas por los tres algoritmos mencionados anteriormente. A este efecto se obtienen algunas medidas objetivas de distancia resultantes de la comparación entre la señal de voz ruidosa o realzada y la señal de voz no contaminada (voz original). En la Tabla.1 se han representado 5 medidas de distancia diferentes: dos medidas temporales, la Relación Señal a Ruido Global (SNR) y la Relación Señal a Ruido Segmentada (SEGSN); y tres medidas espectrales, las distancias Itakura ((ITAKU), Cosh y Cepstrum (CEPST). Sin embargo, las medidas anteriores se han obtenido de forma global sobre una frase completa de un locutor y, consecuentemente, no informan de forma local si un determinado tipo de fonema recibe una mayor o menor distorsión. Así se ha efectuado un estudio trama a trama para evaluar la distorsión y supresión de ruido a lo largo de las distintas tramas correspondientes a tipos de sonidos diferentes.

a) Bajo Nivel de Ruido (SNR=18dB)

Se considera un entorno poco ruidoso donde la señal de voz se ha visto contaminada por la presencia de algo de ruido, originando una SNR de 18dB. Debe remarcarse que el ruido se distribuye de forma uniforme a lo largo de toda una frase de voz. Sin embargo, el nivel energético de la señal de voz puede variar significativamente entre una trama u otra. De este modo, en términos relativos, los sonidos sonoros perciben un nivel de ruido bastante inferior al nivel de ruido apreciado en las tramas correspondientes a sonidos sordos.

A partir de los resultados globales, presentados en la Tabla.1, se puede apreciar que el algoritmo de tercer orden AR3 (Tabla.1.f) logra eliminar casi totalmente el ruido existente tras procesar la primera iteración. Para este mismo nivel de ruido, el algoritmo clásico AR2 (Tabla.1.e) precisa procesar tres iteraciones por trama como mínimo. Los resultados obtenidos con el algoritmo de cuarto orden AR4 también conducen al procesado de una única iteración. Se aprecia claramente como el algoritmo AR3 comporta un ahorro de coste operacional en comparación a los restantes algoritmos (AR2 y AR4). En los test de audición realizados, para las iteraciones óptimas de cada uno de los tres algoritmos evaluados, se aprecia una buena calidad en la señal de voz realzada: el ruido ha desaparecido totalmente y la distorsión causada es tan poco significativa que no produce ninguna percepción de pérdida de inteligibilidad para el oido humano.

Además, se ha realizado un estudio detallado, trama a trama, para evaluar como varia el compromiso existente entre supresión de ruido y distorsión ocasionada. Para los tres algoritmos se puede apreciar un comportamiento común: a medida que se procesan las diferentes iteraciones de una trama de voz, el nivel de ruido disminuye progresivamente hasta quedar enmascarado por el nivel de señal de voz, pero también aparece un efecto negativo, la distorsión, que se manifiesta en forma de desplazamiento de los formantes y/o el efecto de picado de los formantes (formantes más estrechos y valles espectrales más profundos) [4]. Cuando la distorsión se hace más notoria se manifiesta incluso en forma de desdoblamiento de un formante en dos picos espectrales laterales. Así se aprecia una progresiva reducción de ruido, iteración a iteración, hasta alcanzarse una iteración óptima donde, a partir de ésta, el nivel de ruido suprimido es poco significativo y, en cambio, la distorsión introducida en cada iteración posterior es bastante significativa. La diferencia fundamental entre los distintos algoritmos reside en la agresividad de cada uno para afrontar la supresión de ruido y, además, en el grado de distorsión ocasionada. Para este nivel de ruido (SNR=18dB), la zona espectral de baja frecuencia (alrededores del primer formante de la voz) apenas resulta afectada por la presencia del ruido, poniéndose éste de manifiesto en los alrededores del segundo y tercer formante, especialmente en los sonidos sordos de baja energía.

El algoritmo AR3 se muestra bastante agresivo tras procesar únicamente una iteración por trama. Para el caso de los sonidos sonoros parece que la ejecución de la primera iteración es demasiado agresiva para atacar el bajo nivel de ruido presente, y ello comporta distorsiones manifestadas por pequeños desplazamientos de formantes y, en algun caso, se pueden apreciar efectos de picado espectral (especialmente en el primer formante) y algun caso de desdoblamiento. Estos efectos cobran especial relevancia a partir del procesado de la segunda iteración. Si ello fuera posible, se podría afirmar que la iteración óptima está situada antes de la primera iteración. Para los sonidos sordos, donde el nivel relativo de ruido es algo superior, el algoritmo AR3 logra suprimir el ruido presente tras procesar la primera iteración. La distorsión causada es poco notoria, aunque se observan desplazamientos de formantes y pequeños efectos de picado y desdoblamiento. El procesado de más de una iteración por trama comporta la aparición de un efecto de picado espectral mucho más notorio y mayores desplazamientos de formantes. Se podría deducir, en este caso, que la iteración óptima coincide con la primera.

El algoritmo AR2 clásico muestra un comportamiento más conservador: elimina el ruido de una forma más pausada y causa menos distorsión. Para los sonidos sonoros se precisan dos iteraciones para eliminar el ruido

existente y el nivel de distorsión causado es bastante parecido al nivel asociado con la primera iteración de AR3. Para algun fonema vocálico particular, incluso con una sola iteración se logra suprimir el ruido existente. Para los sonidos sordos se precisa el procesado de tres iteraciones por trama para suprimir la mayor parte de ruido y, en algun fonema particular, se precisan incluso cuatro iteraciones.

El algoritmo AR4 muestra un comportamiento intermedio entre los dos anteriores. Para los sonidos sonoros, la iteración óptima se corresponde con la primera iteración. El nivel de distorsión causada es algo inferior, en términos relativos, al que aparece en las iteraciones óptimas de los dos algoritmos anteriores. Para los sonidos sordos, tras la primera iteración todavía se aprecia la presencia de ruido (zona de alta frecuencia), aunque la mayor parte ha sido eliminado. Si se procesa una segunda iteración, el ruido se suprime totalmente pero la distorsión se manifiesta notoriamente, tendiendo a desaparecer algunos formantes debido al efecto de picado asociado con el primer formante.

En resumen, el uso de las estadísticas de tercer orden dotan de tal agresividad al método de Filtrado de Wiener que hacen innecesaria la posible consideración de su versión iterativa. Así, para el margen de niveles de ruido superiores al correspondiente a SNR=15dB, se puede utilizar la técnica clásica de Filtrado de Wiener con estimación AR de tercer orden. Ello conlleva un ahorro considerable en el tiempo de cálculo debido a la ejecución de una única iteración por trama de voz. Si se continua iterando, a partir de la iteración óptima, aparece una apreciable distorsión. En términos estrictos de calidad se podría considerar la elección del algoritmo AR4, pero su menor efecto de distorsión no se pone de manifiesto durante las pruebas de audición. Con la elección del algoritmo AR3 se obtiene, además, un importante ahorro en términos de complejidad.

b) Nivel Medio de Ruido (SNR=9dB)

Para este nivel de ruido el espectro de la señal de voz queda enmascarado totalmente por la presencia de ruido y únicamente la zona del primer formante, durante las tramas sonoras, permanece bastante inalterada por el ruido. De forma global, se aprecia como el algoritmo AR3 (Tabla.1.d) alcanza su iteración óptima durante la primera o segunda iteración, mientras el algoritmo clásico AR2 (Tabla.1.c) precisa como mínimo de tres iteraciones. El algoritmo AR4 muestras unas prestaciones intermedias, precisando dos iteraciones para eliminar el ruido. Las pruebas de audición muestran como los algoritmos AR3 y AR4 permiten eliminar el ruido tras procesar dos iteraciones, con una distorsión muy poco notoria. En cambio, el algoritmo AR2 contiene todavia presencia de ruido tras tres iteraciones y al procesar la cuarta iteración casi desaparece la existencia de ruido pero, en cambio, la distorsión es apreciable en algunos sonidos.

En el estudio detallado por tramas, se aprecia para las tramas sonoras como el algoritmo AR3 precisa únicamente una iteración para eliminar el ruido, e incluso en algunos casos parece como si la iteración óptima estuviera situada un poco antes de esta primera iteración, pues se aprecia efecto picado y desplazamiento en la región del primer formante. También se puede apreciar una mejor recuperación de la zona correspondiente a los formantes superiores tras procesar una iteración de AR3 que al procesar 2 iteraciones de AR2, y consecuentemente origina una menor distorsión. El algoritmo AR4 presenta un comportamiento intermedio: algunas tramas sonoras precisan de una sola iteración mientras que en ótras se precisan 2 iteraciones. Durante las tramas sordas, el algoritmo AR3 alcanza su iteración óptima tras la primera iteración y solamente en algunos casos aislados se precisa una segunda iteración, debido a la persistencia de algo de ruido. En cambio, el algoritmo AR2 precisa como mínimo 3 iteraciones y en algunas tramas la presencia de ruido aun es bastante notoria. Para el algoritmo AR4 se precisan 2 iteraciones en la mayoría de tramas.

A modo de resumen, se puede concluir que todos los algoritmos permiten eliminar este nivel de ruido, aunque algunas tramas sordas procesadas con AR2 contienen todavía un remanente de ruido. En base a los criterios de complejidad y pérdida de inteligibilidad parece más adecuado elegir el algoritmo AR3 para afrontar niveles medios de ruido. La distorsión es un poco superior al supuesto de un entorno poco ruidoso, pero no representa una pérdida de inteligibilidad remarcable para el oido humano.

c) Alto Nivel de Ruido (SNR=0dB)

La situación de señal de voz ruidosa para SNR=0dB corresponde a un nivel crítico de ruido donde la gran mayoría de técnicas de micrófono simple (una sola señal disponible) no obtienen resultados aceptables, tal como se puede apreciar en el caso de segundo orden, donde una mejora gradual y regular se obtiene en las primeras iteraciones y no empieza a saturar hasta superadas unas cuatro iteraciones por trama. Algunos autores proponen soluciones basadas en técnicas multimicrófono para obtener niveles aceptables de calidad e inteligibilidad a la salida, mediante el procesado de varias señales procedentes de distintos micrófonos colocados estratégicamente [3].

Globalmente, el algoritmo AR3 (Tabla.1.b) permite eliminar la mayor parte de ruido tras procesar 3 iteraciones por trama, mientras el algoritmo AR2 (Tabla.1.a) precisa 4 o 5 iteraciones y, además, el nivel remanente de ruido es mucho más notorio. El algoritmo AR4 precisa 5 iteraciones para eliminar casi completamente el ruido. Las pruebas de audición demuestran como el algoritmo AR3 es el único de los tres que logra eliminar casi totalmente

el ruido existente, aunque para estos niveles de ruido aparece una ligera pérdida de inteligibilidad.

El estudio detallado por tramas muestra como el algoritmo AR3 precisa, durante las tramas sonoras, únicamente una iteración para eliminar el ruido, mientras el algoritmo AR2 precisa 2 iteraciones como mínimo. Se aprecia en ambos casos un efecto de picado espectral y desplazamiento de formantes más notorio que en los entornos previamente descritos. El algoritmo AR4 precisa segun las distintas tramas 1 ó 2 iteraciones para eliminar el ruido. Durante las tramas sordas, el algoritmo AR3 precisa como mínimo 2 iteraciones, para alcanzar una supresión de ruido superior a la tercera o cuarta iteración de AR2. Los mayores efectos de picado espectral aparecen en este tipo de tramas sordas correspondientes a este tipo de entornos altamente ruidosos.

Para estos entornos, únicamente el algoritmo AR3 permite afrontar el nivel de ruido existente, mientras se produce una ligera pérdida de inteligibilidad, tras procesar tres iteraciones por trama. Los restantes algoritmos no logran eliminar totalmente el ruido existente y suelen ocasionar una distorsión bastante más notoria.

4. CONCLUSIONES

Se han evaluado las prestaciones de dos algoritmos supresores de ruido (AR3 y AR4), basados en estadísticas de orden superior (HOS) y se han comparado sus prestaciones con las del algoritmo iterativo de Wiener clásico (AR2), basado en la función de segundo orden (autocorrelación). Se ha desarrollado un minucioso estudio, trama a trama, para cada uno de estos algoritmos. En este trabajo se muestra como los algoritmos AR3 y AR4 permiten afrontar entornos de trabajo más ruidosos en relación al algoritmo AR2 clásico. Para distintos entornos ruidosos, el algoritmo basado en los cumulantes de tercer orden (AR3) suele conducir a un mejor compromiso entre supresión de ruido, pérdida de inteligibilidad y complejidad operacional, especialmente en aquellos entornos donde las condiciones de ruido sean más adversas.

5. REFERENCIAS

[1] C.L.Nikias, J.M.Mendel. "Signal Processing with Higher-Order Spectra". IEEE Sig. Proc. Mag. Vol. 10, No. 3, pp 10-37. Julio 1993.

[2] J.S.Lim, A.V.Oppenheim,"All-Pole Modeling of Degraded Speech". IEEE Trans. ASSP, Vol. ASSP-26, No. 3, pp197-210. Junio 1978.

[3] D.Van Compernolle. "DSP Techniques for Speech Enhancement". Proc. ESCA Workshop on Speech Processing in Adverse Conditions, pp. 21-30. Cannes, Francia. 10-13 Noviembre 1992.

[4] E.Masgrau,J.M.Salavedra,A.Moreno,A.Ardanuy. "Speech Enhancement by Adaptive Wiener Filtering based on Cumulant AR Modelling". Proc. ESCA Workshop on Speech Processing in Adverse Conditions, pp 143-146. Cannes, Francia. 10-13 Noviembre 1992.

Tabla 1. *Medidas de distancia para los algoritmos y niveles de ruido siguientes: a) AR2 (autocorrelación) para SNR=0dB; b) AR3 (cumulantes de tercer orden) para SNR=0dB; c) AR2 para SNR=9dB; d) AR3 para SNR=9dB; e) AR2 para SNR=18dB; f) AR3 para SNR=18dB.*

a)	SNR	SEGSN	ITAKU	COSH	CEPST
0 iter.	0.00	0.77	9.57	11.67	12.02
1 iter.	7.36	4.39	8.86	10.43	10.81
2 iter.	8.90	6.03	8.27	9.73	9.66
3 iter.	9.04	6.33	6.73	8.58	8.91
4 iter.	9.13	6.42	5.82	8.07	8.90

b)	SNR	SEGSN	ITAKU	COSH	CEPST
0 iter.	0.00	0.77	9.57	11.67	12.02
1 iter.	7.92	4.86	8.28	9.87	9.91
2 iter.	7.89	5.43	6.03	8.28	8.57
3 iter.	7.97	5.74	5.31	7.68	8.15
4 iter.	7.84	5.92	5.11	7.63	8.29

c)	SNR	SEGSN	ITAKU	COSH	CEPST
0 iter.	9.00	8.07	8.27	9.92	10.51
1 iter.	14.46	10.13	7.15	8.61	9.17
2 iter.	15.74	11.47	6.13	7.58	7.93
3 iter.	15.91	11.68	4.42	6.30	7.05
4 iter.	15.86	11.69	3.58	5.81	7.01

d)	SNR	SEGSN	ITAKU	COSH	CEPST
0 iter	9.00	8.07	8.28	9.92	10.51
1 iter	14.50	10.77	4.64	6.85	7.23
2 iter	14.17	10.90	4.26	6.39	7.15
3 iter	14.07	10.71	3.99	6.07	7.01
4 iter	13.46	10.63	3.77	5.93	6.98

e)	SNR	SEGSN	ITAKU	COSH	CEPST
0 iter	18.00	13.41	6.33	7.89	8.52
1 iter	21.76	16.74	4.90	6.43	6.96
2 iter	22.35	17.47	3.75	5.42	5.75
3 iter	22.29	17.52	2.49	4.59	5.21
4 iter	22.04	17.33	2.07	4.36	5.27

f)	SNR	SEGSN	ITAKU	COSH	CEPST
0 iter	18.00	13.41	6.33	7.89	8.52
1 iter	21.18	16.78	2.68	4.89	5.84
2 iter	20.26	16.12	2.55	4.83	6.22
3 iter	18.78	15.40	2.65	4.92	6.42
4 iter	19.04	15.37	2.67	4.96	7.56