using the original MBE pitch analyser, but these errors are effectively removed by using the improved pitch analyser (unbroken). We also found that the improved pitch detection algorithm performed very well, even without pitch tracking, therefore the algorithmic coding delay was shortened to 32ms. A full-duplex MBELP coder employing the improved pitch detection algorithm was successfully implemented in real time on a single C31 DSP, and pitch estimation requires only 16% of the processor resource.

**Conclusion:** We propose a new error measure for spectrum matching in MBE pitch analysis of speech. By applying a corrective measure to the original error measure for pitch estimation, gross pitch errors can be effectively removed. The corrective measure is based on a sum-of-product formula which facilitates fast searching of the optimum pitch period by using partial-sum comparisons.

---

**Motion estimation with object based regularisation**

S. Panis and J.P. Cosmas

*Indexing terms: Motion estimation, Dynamic programming*

A dynamic programming based matching method for motion estimation, that optimises a Bayesian maximum likelihood function in a 3-D optimisation space, is presented. The Bayesian function consists of a matching cost and an object based 2-D regularisation cost. The method gives results more accurate than block-based matching since the motion boundaries are close to the actual object boundaries.

**Introduction:** A dynamic programming-based matching method for motion estimation is presented which optimises a Bayesian maximum likelihood function that consists of two parts: a matching cost and a 2-D regularisation cost that enforces the motion vectors to be a monotonic function only within objects, thereby allowing an edge preserving transition between objects and hence object-based regularisation. A 3-D optimisation space is used to accommodate the two degrees of motion freedom.

---

**Cost function:** The total cost function for dynamic programming is composed of a combination of a normalised matching cost and a regularisation cost. The normalised matching cost \( C_{MC} \) is determined by the quality of matching. The regularisation cost consists of two parts. One part enforces the vectors across a surface to be a monotonic function (monotonicity constraint) and the other part controls the strength of the monotonicity function so that its effect is dependent on the luminance variations and hence acts strongly inside objects but very weakly at the edges. Cox et al. [2] claim that enforcement of regularisation may harm the results at the edges where they are most accurate owing to the rich texture. Geiger et al. [1] state that regularisation is necessary in order to obtain accurate results even where matching is poor, e.g., within objects with poor texture. The cost function presented here takes account of both points of view. By reducing the effect of the monotonicity cost at the edges, the dynamic programming can use the matching cost \( C_{MC} \) as the main decision cost. Since matching near edges is usually better than within a surface, the \( C_{MC} \) has high confidence at the edges and this leads the program to a correct decision without regularisation influence. A typical problem occurs at two consecutive pixels that belong to two different objects with large motion vector difference. If no consideration for edges was there, the monotonicity constraint alone would impose vector uniformity, shooting the total cost high and resulting in very large punishment. The area would have been interpreted as unmatched.

The regularisation cost function is defined as

\[
f(x,s) = \frac{\mu \sqrt{\| x \| + \| z \|}}{\delta y} \quad x \geq 0, s \geq 0anumber{1}
\]

where \( x \) is the amount of motion vector jump between consecutive pixels of the raster line and \( s \) is the normalised luminance gradient extracted using a Sobel edge detector. The plot of the cost function is shown in Fig. 3. The weights \( \mu, \beta, \alpha \) are empirically determined constants. For \( s \leq 1 \) i.e. when there is practically no edge, the function is punishing only motion vector jumps and hence acting purely as a monotonicity constraint [1]. For \( s > 1 \), i.e. when there is a strong edge, the effectiveness of the motion vector jump function is reduced to allow a jump without strong punishment. According to Fig. 3, as \( s \) increases the smoothness criterion is pulled down to allow jumps between interpolated pixels. The total cost \( T(j,i) \) is calculated with the recursive function

\[
T_{\mu}(j,i) = C_{NMC}(j,i) + f(x_{a},s) + f(x_{b},s)
\]

\[
+ T_{\mu-1}(j,\text{horizontal}_{:}i) \quad \text{for } n > 1
\]

\[
T_{\mu}(j,i) = C_{NMC}(j,i) \quad \text{for } n = 1
\]

\[
(2)
\]

where \( x_{a} \) is the horizontal motion jump between the current candidate horizontal motion vector and the horizontal motion vector of the previous column, \( x_{b} \) is the respective vertical motion jump, \( n \) is the current column number, \( j \) is the candidate horizontal motion vector and \( i \) is the candidate vertical motion vector. \( \text{horizontal}_{:}i \) is the best predecessor vector.
of the motion vectors, block artefacts and the mosquito effect are not present. The proposed algorithm was used in the object based coder of [4] and experiments showed that errors owing to motion compensation were considerably lower than when block based matching was used, with the performance of the coder being better than that of MPEG-2.

Acknowledgments: This work was done within the framework of RACE (Research on Advanced Communications in Europe) under contract R2045 ‘DISTIMA’ (DIGitial STereoscopic IMaging and Applications).

© IEE 1996
Electronics Letters Online No: 19960598
S. Panis (Siemens AG, ZFE T SN 2, 81730 Munich, Germany)
J.P. Cosmas (Queen Mary and Westfield College, Electronic Engineering Department, Mile End Road, London E1 4NS, United Kingdom)

References
3 CCIR 601 format sequences shot within the EU project RACE DISTIMA

Trellis decoding of combined diversity-coding scheme (MLSD) for fading channels

K. Mohamed-Pour and P.G. Farrell

Indexing terms: Trellis coded modulation, Fading, Diversity reception

The performance of a receiver using a combination of soft-decision decoding and diversity reception is investigated for nonselective multipath Rayleigh fading channels. A new scheme for soft-diversity, soft-decision detection, maximum likelihood selection and decoding (MLSD), is introduced, in which decisions on the diversity channels and decoding are carried out simultaneously by using a trellis and the Viterbi algorithm.

Introduction: Diversity reception is one way to improve the reliability of communication without increasing either the transmitted power or the bandwidth. Coding is another advantageous way which can be used for power-limited and/or bandlimited channels [1]. In this Letter a combination of diversity detection and decoding, especially in the soft-decision decoding case, is introduced which improves the performance of the transmission of data over a multipath fading channel, compared with separate diversity detection and decoding.

Diversity with soft decision decoding: Supposing an L-diversity scheme, the system is modelled as L independent multipath Rayleigh channels corrupted by additive white Gaussian noise n(t). The L receivers are assumed to employ matched filters to get the maximum signal/noise ratio at sampling times. Values \( r_j, j = 1, 2, \ldots, n \), are the unquantised samples of the outputs. These samples are then quantised to Q levels and then accumulated in an \( L \times n \) buffer matrix as \( \{ y_{ij} \} \) values which can be used in a diversity combination scheme, involving a code of length n.

Soft-diversity decoding' refers to any decoding process on the soft values of the outputs in a diversity system which may involve different combination techniques. Two such techniques are: