EXTRACTION OF INDEPENDENT COMPONENTS FROM HYBRID MIXTURE: KUICNET LEARNING ALGORITHM AND APPLICATIONS

S. Y. Kung
Princeton University

Cristina Mejuto
University of La Coruna

ABSTRACT

A *hybrid mixture* is a mixture of supergaussian, gaussian, and subgaussian independent components(ICs). This paper addresses extraction of ICs from a hybrid mixture. There are two kinds of (single-output vs. all-outputs) kurtosis function to be considered as a contrast function. We advocate the former approach due to its (1) simple and closed-form analysis, and (2) numerical convergence and computational saving. Via this approach, all (and only) the positive local maxima (resp. negative local minima) can yield supergaussian (resp. subgaussian) ICs from any mixture[5]. We also propose a network algorithm, Kurtosis-based Independent Component Network(KuicNet), for recursively extracting ICs. Numerical and convergence properties are analyzed and several application examples demonstrated.

1. INTRODUCTION

Independent component analysis(ICA) extracts components with higher-order statistical independence. It is to find linear feature extraction such that the extracted components are as independent as possible. It has found potential applications in blind source separation e.g. "cocktail party" problem, sensor array processing, interference or noise reduction/removal, and finding minimum entropy code, etc. The work has prompted a lot of interests in neural network community thanks to notion of an information-maximization approach to blind separation and blind convolution[1, 2, 3].

A *nontrivial* **hybrid mixture** is one containing supergaussian *as well as* subgaussian components. The paper aims at addressing the extrema property of kurtosis, thus establishing an effective contrast function and a KuicNet learning rule for IC extraction from such *hybrid mixtures*.

1.1. Single-Output Kurtosis Contrast Function

Throughout the paper, all the source signals $\{s_i\}$ are assumed to be mutually independent (up to the 4-th order):

$$E[s_i] = 0, \qquad E[s_i^2] = 1, \quad \text{and}$$

$$E[s_i^k s_j^l s_m^p s_n^q] = E[s_i^k]E[s_j^l]E[s_m^p]E[s_n^q] \quad \forall \quad k+l+p+q \leq 4$$

Given a vector $\mathbf{x}$ consists of N observation processes each being a linear combination of the source signals: $\mathbf{x} = \mathbf{A}\mathbf{s}$, where $\mathbf{A}$ is an unknown $N \times N$ (mixer) matrix. The main problem for single-output ICA is to design a contrast function whose maximization would yield a scalar process $y(t)$, $y = \mathbf{m}^T\mathbf{x}(t)$, so that $y(t)$ extracts one of the N independent components. (We also denote $y(t) = \mathbf{b}^T\mathbf{s}(t)$, where $\mathbf{b}^T = \mathbf{m}^T\mathbf{A}$.)

Figure 1.1. depicts a network configuration (with one output y). for the extraction of the primary component under a given criterion. $y = \mathbf{m}^T x$ where $\mathbf{x}(t)$ is zero-mean vector process, i.e. $\mathbf{x}(t) \in R^N$, an N-dimensional vector space.

$$\mathbf{s}(t) \rightarrow \mathbf{A} \rightarrow \mathbf{x}(t) \rightarrow \mathbf{m}^T \rightarrow y(t)$$

(a) The original observation space $\mathbf{x}$.

$$\mathbf{s}(t) \rightarrow \mathbf{A} \rightarrow \mathbf{x}(t) \rightarrow \mathbf{P} \rightarrow \mathbf{v}(t) \rightarrow \mathbf{w}^T \rightarrow y(t)$$

(b) Prewhitening process.

$$\mathbf{s}(t) \rightarrow \mathbf{\Gamma} \rightarrow \mathbf{v}(t) \rightarrow \mathbf{w}^T \rightarrow y(t)$$

(c) The reproduced observation space $\mathbf{v}$.

Figure 1. (a) The mixing process is represented by an unknown matrix A. (b) Prewhitening procedure. (c) Ignoring x and operating on the reproduced space v proves to be numerical advantageous.

Kurtosis: The **kurtosis** of a process y is defined as

$$k(y) = f(y) - 3, \qquad \text{where} \qquad f(y) = \frac{E[y^4]}{E[y^2]^2}$$

Note $k(y)$ and $f(y)$ are scale-invariant, i.e. $f(\alpha y) = f(y)$.

A scale-invariant functional $\phi(y)$ is a valid contrast function for separating the sources $\{s_i, i = 1, \cdots, N\}$ if

$$min_i\{\phi(s_i)\} < \phi(\sum_i \alpha_i s_i) < max_i\{\phi(s_i)\} \qquad (1)$$

for at least two nonzero coefficients $\{\alpha_i\}$[5].

1.2. Numerical Advantages

- Just like PCA extraction, it was proposed to extract one IC at one time[4]. It is especially *advantageous for the hybrid mixture*, because the single-component extraction effectively circumvents the effect of the mixed positive and negative kurtosis - which has plagued the "all-outputs" contrast functions.
- In [4], a squared-kurtosis function was adopted as the contrast function. From the numerical perspective, it is **not** advisable to use the squared-kurtosis as contrast function as its corresponding adaptive algorithm **necessitates** the difficult task of estimating $k(y)$. It is numerically superior to work with kurtosis, instead of

Cases	Extrema for IC	Non-IC Extrema
super+sub	N extrema	None
super+G & sub+G	N extrema	None
super+sub+G	$N-1$ extrema	None
all-super	N maxima(+)	2^{N-1} minima(+)
all-sub	N minima(-)	2^{N-1} maxima(-)

Table 1. The extrema of the kurtosis function.

squared-kurtosis, since it allows us to circumvent the need of estimating k(y) - and the associated numerical hazard.

- During maximizing Eq. 5 via computing gradient w.r.t. $\mathbf{m}$, the division (due to the denominator in Eq. 5) incurs severe numerical difficulty. This can be circumvented altogether by working on a reproduced space $\mathbf{v}$ derived by pre-whitening(Section 3.), which has been a popular technique in the ICA literatures. Thereafter, a gradient method leads to our KuicNet learning rule, cf. Section 4.
- Once one IC is extracted, a deflation algorithm (which is well-established in numerical analysis literature) may be applied to remove that IC from the mixture signals. Thereafter, the other components may be extracted by a recursive procedure, cf. Section 4.

2. EXTREMA OF KURTOSIS FUNCTION

We shall from now on - without loss of generality - reorder the source signals according to the values of the kurtosis:

$$k(s_1) \geq k(s_2) \geq .....k(s_N) \tag{2}$$

Let p denote the number of the supergaussian and m that of the subgaussian components. In general, $p+m \leq N$. In the absence of zero-kurtosis processes[1] then $N = p+m$.

Theorem 2..1 *The kurtosis function as a function of* $\mathbf{m}$, $k(y) = k(y(\mathbf{m}))$, *has the following extrema:*

- *If* $k_1 \geq 0$ **and** $k_N \leq 0$, *i.e. assuming a (nontrivial) hybrid mixture, then the kurtosis function* $k(y)$ *(and* $f(y)$*) meets the condition Eq. 1 on contrast function. For this case,* **there are** p **maxima and** m **minima.** *The output* y *extracts a supergaussian IC (resp. a subgaussian IC)* **if and only if** *a local maximum (resp. minimum) is reached. The extreme value equals the kurtosis of the extracted independent component. (See Figures 2(a)(b)(c).)*
- *For the homogeneous mixture cases, assuming* $k_N > 0$ *i.e. all-supergaussian, then the maxima of* $k(y)$ *yield ICs, but not the minima. (See Figure 2(d).) Similar arguments hold for teh case* $k_1 < 0$.
- *For all the cases,* **all** *the non-negative maxima and non-positive minima extract ICs and* **only** *these extrema (i.e. neither negative maxima nor positive minima) yield any pure ICs. (See Figure 2 and Table 1.)*

The theorem follows directly the extrema property of kurtosis function as established in [5], illustrating (1) the number of extrema (modulo a scaling factor), (2) the locations of the extrema, and (3) the corresponding extreme values.

As exemplified in Figure 2, there are 3 ICs. If s_1 and s_2 are supergaussian while s_3 either subgaussian (i.e. case "super+sub") or gaussian (i.e. case "super+G"), then according to Table 1 there are $N = 3$ extrema. The two maxima of $k(y)$ yield s_1 or s_2, while the only minimum yields s_3, cf. Figures 2 (a) and (c).

[1] A gaussian component must have zero kurtosis, but a zero-kurtosis process needs not to be gaussian.

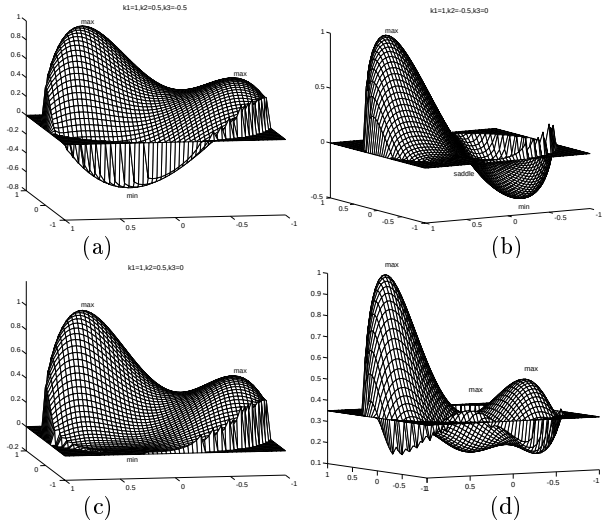

Figure 2. The extrema w.r.t. the 2 free variables in the normalized m space for 3 independent sources (here 2 = 3-1). Cases: (a) super+sub, (b) super+sub+G, (c) super+G, and (d) all-super.

Figures 2(d) shows a total of $N + 2^{N-1} = 7$ extrema for the case of 3 "all-supergaussian" ICs. The 3 positive-valued maxima yield 3 ICs, while the 4 (positive-valued) minima yield mixed signals. (Note only 3 of the 4 minima are visible in the figure.)

For the all-supergaussian case ($k_N > 0$), there are $p = N$ local maxima (with the extreme values being $k_l, \forall l \leq p$) and 2^{N-1} (positive equal-valued) minima. See Figure 2 (d). The output y extracts one of the p supergaussian ICs **if and only if** a local maximum of $k(y(\mathbf{m}))$ is reached.

The local minima, on the other hand yield mixed outputs:

$$y = \sum_j^N b_j s_j, \qquad \text{where} \qquad b_j = \pm\sqrt{\frac{k_j^{-1}}{\sum_{l=1}^N k_l^{-1}}} \tag{3}$$

There are only 2^{N-1} local maxima, modulo the scale invariance (i.e. with scale factor = -1), although 2^N possible combinations of different signs exist. The extreme values are all equal to the homogeneous mean: (Fig. 2(d)).

$$k(y) = [\ \sum_{l=1}^N k_l^{-1}\ ]^{-1} \quad > \quad 0 \tag{4}$$

3. REPRODUCED SPACE VIA PRE-WHITENING PROCESS

To find a solution $y = \mathbf{m}^T\mathbf{x}$ to (locally) maximize

$$\mathbf{MAX_m} \quad \phi(y) = \mathbf{MAX_m} \quad \frac{E[(\mathbf{m}^T\mathbf{x})^4]}{E[(\mathbf{m}^T\mathbf{x})^2]^2}, \tag{5}$$

then the division by $E[y^2] = E[(\mathbf{m}^T\mathbf{x})^2]$ is vulnerable to numerical mishap and thus must be avoided if possible. A simple way out is by creating a reproduced space $\mathbf{v}$. See Figure 1.1.(c).

The reproduced space is generated by the following steps:

- First, we generate a normalized and pre-whitened process $\mathbf{v} = \mathbf{P}\ \mathbf{x}$. Each element of $\mathbf{v}$ corresponds to one of the (conventional) principal components with variance = 1.

- Note that $y = \mathbf{w}^T\mathbf{v} = \mathbf{w}^T\mathbf{P}\mathbf{x}$,
  So $\mathbf{m} = \mathbf{P}^T\mathbf{w}$. From now on, we work with the new representation of y, cf. Figure 1.1.(c):

$$y = \mathbf{w}^T\mathbf{v}$$

The key advantage of operating in the space $\mathbf{v}$, is that

$$E[y^2] = E[(\mathbf{w}^T\mathbf{v})^2] = \|\mathbf{w}\|^2 \qquad (\quad = \|\mathbf{b}\|^2\ ) \quad (6)$$

has the appearance of a function of only $\mathbf{w}$ - instead of $\mathbf{v}$ and $\mathbf{w}$. (This attribute is also enjoyed by the source space $\mathbf{s}$, i.e. $E[y^2] = \|\mathbf{b}\|^2$.) In this sense, $\mathbf{v}$ space exhibits a similar property to the source space $\mathbf{s}$. This is why $\mathbf{v}$ is called a **reproduced space** of $\mathbf{s}$.

Numerical Procedure for Reproducing v-space Assuming that a total of M observation samples are available, here we describe a numerically efficient procedure for deriving $\mathbf{v}$(t) and $\mathbf{V}$, from $\mathbf{x}$(t) and $\mathbf{X}$, where

$$\begin{aligned}\mathbf{X} &= [\mathbf{x}(1)|\mathbf{x}(2)|\cdots|\mathbf{x}(M)]\\ \mathbf{V} &= [\mathbf{v}(1)|\mathbf{v}(2)|\cdots|\mathbf{v}(M)]\end{aligned}$$

1. Compute a covariance matrix of $\mathbf{x}$

$$\mathbf{R}_\mathbf{x} = \mathbf{X}\mathbf{X}^T$$

2. Apply SVD to

$$\mathbf{X}\mathbf{X}^T = \mathbf{U}\Sigma^2\mathbf{U}^T$$

3. Compute $\mathbf{V}$ as

$$\mathbf{v}(t) = \Sigma^{-1}\mathbf{U}^T\mathbf{x}(t),$$

Thus

$$\mathbf{V} = \Sigma^{-1}\mathbf{U}^T\mathbf{X}$$

So the covariance matrix of $\mathbf{v}$

$$\mathbf{R}_\mathbf{v} = \mathbf{V}\mathbf{V}^T = I$$

To verify the condition stated in Eq. 6, we note

$$E[y^2] = \mathbf{w}^T\mathbf{V}\mathbf{V}^T\mathbf{w} = \|\mathbf{w}\|^2$$

Note also that

$$\mathbf{v}(t) = \mathbf{P}\mathbf{x}(t) = \mathbf{P}\mathbf{A}\mathbf{s}(t) = \mathbf{,}\ \mathbf{s}(t)$$

and

$$\mathbf{w} = \mathbf{,}\ \mathbf{b}, \qquad \mathbf{b} = \mathbf{,}^T\mathbf{w}$$

where $\mathbf{,} = \mathbf{P}\mathbf{A}$ is a unitary matrix. (Proof: $I = \mathbf{V}\mathbf{V}^T = \mathbf{,}\ \mathbf{S}\mathbf{S}^T\mathbf{,}^T = \mathbf{,}\ \mathbf{,}^T$.)

The ultimate goal is to find an inverse (i.e. de-mixing) matrix $\mathbf{W} = \mathbf{,}^{-1} = \mathbf{,}^T$, (here we ignore permutation and sign). So $\mathbf{W}$ has to be unitary too.

4. KUICNET LEARNING ALGORITHMS

It is numerically advantageous in working on the reproduced space $\mathbf{v}$ instead of $\mathbf{x}$, we have:

$$\phi(y(t)) = E[\frac{(\mathbf{w}^T\mathbf{v}(t))^4}{\|\mathbf{w}\|^4}] = E[g(t)]$$

where $g(t) \equiv \frac{(\mathbf{w}^T\mathbf{v}(t))^4}{\|\mathbf{w}\|^4}$. To derive a **data adaptive learning scheme**, we apply the gradient:

$$\nabla_\mathbf{w} g(t) = 4(\mathbf{v}\frac{y^3}{\|\mathbf{w}\|^4} - \mathbf{w}\frac{y(t)^4}{\|\mathbf{w}\|^6})$$

Since the kurtosis is scale-invariant (and so is g(t)), without loss of generality, we impose $\|\mathbf{w}\| = 1$ to obtain:

$$\nabla_\mathbf{w} g(t) = 4(\mathbf{v}y(t)^3 - \mathbf{w}y(t)^4) \quad (7)$$

This leads to the following:

Algorithm 4..1 (KuicNet Learning Rule)
The KuicNet learning rule to extract a supergaussian component is

$$\Delta\mathbf{w}(t) = +\beta[\mathbf{v}(t)y^3(t) - \mathbf{w}(t)y(t)^4] \quad (8)$$

where β is a small positive learning rate.
Assuming that $s_1 > 0$, then when the above learning rule converges (to a local maximum), the extracted output is a supergaussian source component.
Note: *To extract a subgaussian component (assuming $s_N < 0$), (1) $+\beta$ is replaced by $-\beta$, and (2) $\mathbf{w}(t)$ should be constantly re-normalized during the iterations[5].* ■

Proof: The convergence occurs when and only when (1)

$$E[\nabla_\mathbf{w} g(t)] = 4E[(\mathbf{v}y(t)^3 - \mathbf{w}y(t)^4)] = 0 \quad (9)$$

and (2) the Hessian matrix $\mathbf{H}_\mathbf{w}$) is semi-negative-definite. Pre-multiply Eq. 9 by $\mathbf{,}^T$,

$$0 = \mathbf{,}^T E[(\mathbf{v}y(t)^3 - \mathbf{w}y(t)^4)] = E[(\mathbf{s}y(t)^3 - \mathbf{b}y(t)^4)]$$

Examining its j-th element,

$$0 = b_j(b_j^2k_j - \sum_{l=1}^{N} b_l^4k_l) \quad (10)$$

This leads to the following solution

$$\mathbf{b} = [0,\cdots 0,\pm 1, \quad 0\cdots 0]^T \quad (11)$$

which has all zeros except its j-th element, $j \le p$, and is a local maximum of

$$f(y) = \frac{E[y^4]}{E[y^2]^2} = \sum_{j=1}^{N}\frac{b_j^4}{\|\mathbf{b}\|^4}k_j + 3 = \sum_{j=1}^{N} b_j^4k_j + 3$$

under the constraint $||\mathbf{b}||^2 = ||\mathbf{w}||^2 = 1$.
By the well-known transformation:

$$\nabla_w f(y) = \mathbf{,}\ \nabla_b f(y) \quad \text{and} \quad \mathbf{H}_\mathbf{w} = \mathbf{,}\ \mathbf{H}_\mathbf{b}\mathbf{,}^T$$

since $\mathbf{H}_\mathbf{b}$ is semi-negative-definite, then so is $\mathbf{H}_\mathbf{w}$ (cf. [5]). So we conclude that the solution(s) for Eq. 8 is $\mathbf{w} = \mathbf{,}\ \mathbf{b}$, is indeed a local stable maximum. It follows that the extracted output

$$y = \mathbf{w}^T\mathbf{v} = \mathbf{b}^T\mathbf{,}^T\mathbf{,}\ \mathbf{s} = \mathbf{b}^T\mathbf{s} = \pm s_j$$

yields a supergaussian component.

Algorithm 4..2 (KuicNet for Extracting All ICs)
Three key steps are in the KuicNet Procedure:

1. **Extraction of a supergaussian component:**
   The following learning rule may be adopted for extracting a supergaussian IC (use "+" rule) or, respectively, a subgaussian IC (use "−" rule). The initial weight $\mathbf{w}(0)$ can be any normalized random vector.

$$\Delta\mathbf{w}(t) = \pm\beta[\mathbf{v}(t)y^3(t) - \mathbf{w}(t)y(t)^4] \quad (12)$$

   Numerically, it helps to apply re-normalization, cf. Section 4., The trained vector $\mathbf{w}$ will converge to extract $y = \mathbf{w}^T\mathbf{v}$ as one of the supergaussian (resp. subgaussian) ICs. For completeness, the first column of the de-mixing matrix $\mathbf{M}$ is $\mathbf{m}_1 = \mathbf{P}^T\mathbf{w}$.

2. **Deflation Procedure**
The deflation involves the removal of the extracted IC from the reproduced space $\mathbf{v}(t)$*, resulting in a newer reproduced space* $\mathbf{V'}$*, formed by a set of* $N-1$ *new signals:* $\mathbf{v}'(t)$*. Assuming (WLOG) that* $s_1 = \pm \mathbf{w}^T\mathbf{v}$ *is the IC just extracted which is to be removed.*
To find the subspace $\mathbf{W}_{orth}$ *orthogonal to* $\mathbf{w}$*, we apply SVD to* $I - \mathbf{w}\mathbf{w}^T =$

$$[\ \mathbf{W}_{orth}\ |\ \mathbf{w}\ ] \begin{bmatrix} 1 & & & \\ & \ddots & & \\ & & 1 & \\ & & & 0 \end{bmatrix} \left[\begin{array}{c} \mathbf{W}_{orth}^T \\ \hline \mathbf{w}^T \end{array}\right]$$

where $\mathbf{W}_{orth}$ *is an* $N \times (N-1)$ *matrix, corresponding to the (N-1) singular vectors. By the SVD definition,* $[\mathbf{w}|\mathbf{W}_{orth}]$ *forms a unitary matrix. Obviously,* $\mathbf{V}' = \mathbf{W}_{orth}\mathbf{V}$ *has* $N-1$ *rows, each row is a linear combination of* $\{s_2, s_3, ..., s_N\}$*. (Thus, the extracted output* $y(t)$ *should be independent of* $\mathbf{v}'(t)$ *- a fact may be adopted for our on-line verification.)*
Now the recursion is basically all set. For example, if $\mathbf{w}'$ *yields a second IC, then the second column of* $\mathbf{M}$ *can be back-tracked as*

$$\mathbf{m}_2 = \mathbf{P}^T\mathbf{W}_{orth}\mathbf{w}'$$

So can other components $\{\mathbf{m}_j, j > 2\}$ *be derived.*

3. **Termination Rule:** *(Mode-Switching Rule)*
The procedure terminates, when the output y consistently yields a negative (resp. positive) kurtosis. ■

5. SIMULATION RESULTS

We have performed two experiments each with three sources, with the deflation procedure implemented, and the KuicNet successfully recovered the source signals.

Experiment 1: Mixtures of speech signal and noise
Two speech signals are corrupted by a subgaussian interference noise, with very high noise-signal-ratio. By listening to the actual sounds, as well as inspecting the waveforms depicted in Figure 3, we conclude that KuicNet can recover (two) very clear speech signals from (three) almost non-intelligible sounds.

Experiment 2: Mixture of image/speech signals
As shown in Figure 4, the KuicNet successfully recover one (supergaussian) speech signal and two (subgaussian) images from three multi-media mixtures.

Acknowledgement: The authors thank Prof. Luis Castedo, Y.K. Chen, and Hua Lin for their invaluable inputs. The work was supported in part by Mitsubishi (MEITCA) and Xunta de Galicia (XUGA 10502A96) and CICYT (TIC 96-0500-C10-02).

REFERENCES

[1] Jean-Francois Cardoso, "Source separation using higher order moments" Proceeding, ICASSP89, Glasgow, U.K. 1989.

[2] P. Comon, "Independent Component Analysis, A new concept" Signal Processing, Vol. 36, pp. 287-314, 1994.

[3] A. J. Bell and T. J. Sejnowski, "An information-maximization approach to blind separation and blind convolution, Neural Computation", Vol. 7, MIT Press,1995.

[4] N. Delfosse and P. Loubaton, "Adaptive blind separation of independent sources: a deflation approach" Signal Processing, Vol. 45, pp. 59-83, 1994.

[5] S.Y. Kung, "Independent Component Analysis in Hybrid Mixture: KuicNet Learning Algorithm and Numerical Analysis", Proceedings, Inter. Sympo. on Multimedia Information Processing, pp. 368-381, Taipei, Dec. 1997. (Full version submitted to IEEE Trans. on Signal Processing.)

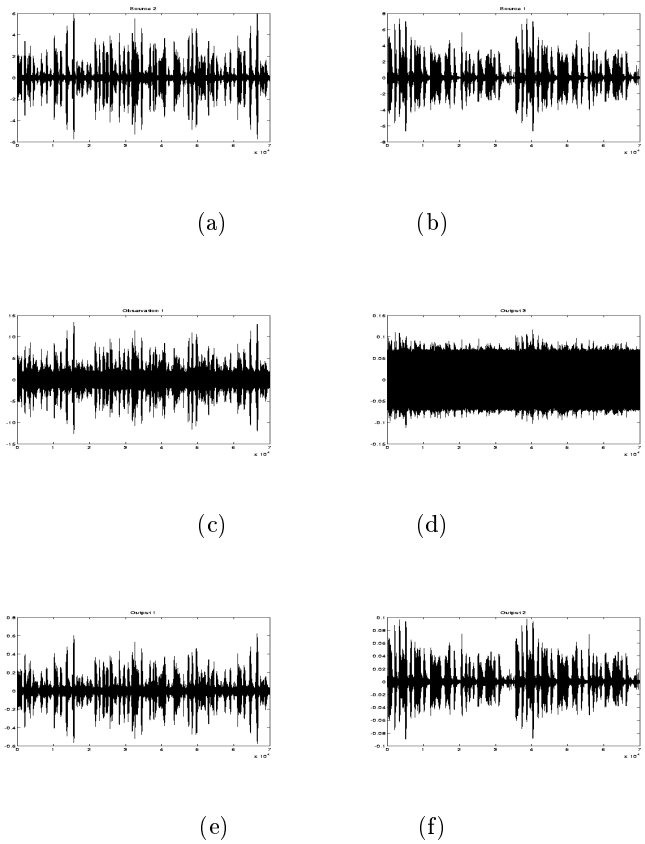

Figure 3. (a) (b) Original speech signals (c) one of the three hybrid mixtures - corrupted by uniform random noise; (d) recovered noise; (e) and (f) recovered and (well) separated speech signals

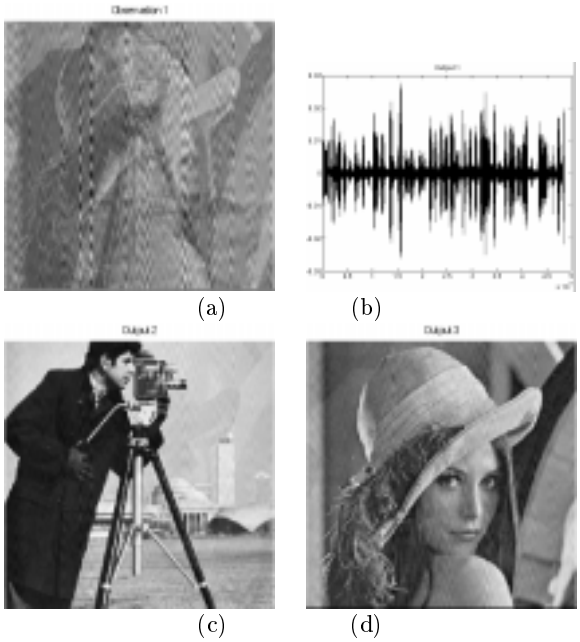

Figure 4. (a) One of the three hybrid mixtures. (b) recovered speech signal (c) and (d) recovered images