Markovian Blind Image Separation

Shahram Hosseini[1], Rima Guidara[1], Yannick Deville[1], and Christian Jutten[2]

[1] Laboratoire Astrophysique de Toulouse-Tarbes (LATT),
Observatoire Midi-Pyrénées - Université Paul Sabatier,
14 Avenue Edouard Belin, 31400 Toulouse, France
{shosseini, rguidara, ydeville}@ast.obs-mip.fr
[2] Laboratoire des Images et des Signaux (LIS),
UMR CNRS-INPG-UJF, Grenoble, France
Christian.Jutten@inpg.fr

Abstract. We recently proposed a markovian image separation method. The proposed algorithm is however very time consuming so that it cannot be applied to large-size real-world images. In this paper, we propose two major modifications *i.e.* utilization of a low-cost parametric score function estimator and derivation of a modified equivariant version of Newton-Raphson algorithm for solving the estimating equations. These modifications make the algorithm much faster and allow us to perform more experiments with artificial and real data which are presented in the paper.

1 Introduction

We recently proposed [1] a quasi-efficient Maximum Likelihood (ML) approach for blindly separating mixtures of temporally correlated, mono-dimensional independent sources where a Markov model was used to simplify the joint Probability Density Functions (PDF) of successive samples of each source. This approach exploits both source non-gaussianity and autocorrelation in a quasi-optimal manner.

In [2], we extended this idea to bi-dimensional sources (in particular images), where the spatial autocorrelation of each source was described using a second-order Markov Random Field (MRF). The idea of using MRF for image separation has recently been exploited by other authors [3], where the source PDF are supposed to be known, and are used to choose the Gibbs priors. In [2], however, we made no assumption about the source PDF so that the method remains quasi-efficient whatever the source distributions. The first experimental results reported in [2] confirmed the better performance of our method with respect to the ML methods which ignore the source autocorrelation [4] and the autocorrelation-based methods which ignore the source non-gaussianity [5], [6].

The algorithm used in [2] is however very slow: its implementation requires the estimation of some 5-dimensional conditional score functions using a non-parametric estimator and the maximization of a likelihood function using a time

J. Rosca et al. (Eds.): ICA 2006, LNCS 3889, pp. 106–114, 2006.

consuming gradient method. In the present paper, we propose a parametric polynomial estimator of the conditional score functions which is much faster than the non-parametric estimator. We also derive a modified equivariant Newton-Raphson algorithm which considerably reduces the computational cost of the optimization procedure. Using this fast algorithm, we performed more simulations with artificial and real-world data to compare our method with classical approaches.

2 ML Method for Separating Markovian Images

Assume we have $N = N_1 \times N_2$ samples of a K-dimensional vector $\mathbf{x}(n_1, n_2)$ resulting from a linear transformation $\mathbf{x}(n_1, n_2) = \mathbf{A}\mathbf{s}(n_1, n_2)$, where $\mathbf{s}(n_1, n_2)$ is the vector of independent image sources $s_i(n_1, n_2)$, each one of dimension $N_1 \times N_2$ and possibly spatially autocorrelated, and $\mathbf{A}$ is a $K \times K$ invertible matrix. Our objective is to estimate the separating matrix $\mathbf{B} = \mathbf{A}^{-1}$ up to a diagonal matrix and a permutation matrix.

The ML method consists in maximizing the joint PDF of all the samples of all the components of the vector $\mathbf{x}$ (all the observations), with respect to the separating matrix $\mathbf{B}$. We denote this PDF

$$f_{\mathbf{x}}(x_1(1,1), \cdots, x_K(1,1), \cdots, x_1(N_1,N_2), \cdots, x_K(N_1,N_2)) \tag{1}$$

Under the assumption of independence of the sources, this function is equal to

$$\left(\frac{1}{|\det(\mathbf{B}^{-1})|}\right)^N \prod_{i=1}^{K} f_{s_i}(s_i(1,1), \cdots, s_i(N_1,N_2)) \tag{2}$$

where $f_{s_i}(.)$ represents the joint PDF of N samples of the source s_i. Each joint PDF can be decomposed using Bayes rule in many different manners following different sweeping trajectories within the image corresponding to source s_i (Fig. 1). These schemes being essentially equivalent, we chose the horizontal sweeping. Then, the joint PDF of source s_i can be decomposed using Bayes rule to obtain

$$\begin{aligned}
&f_{s_i}(s_i(1,1)) f_{s_i}(s_i(1,2)|s_i(1,1)) f_{s_i}(s_i(1,3)|s_i(1,2), s_i(1,1)) \cdots\cdots \\
&f_{s_i}(s_i(1,N_2)|s_i(1,N_2-1), \cdots, s_i(1,1)) f_{s_i}(s_i(2,1)|s_i(1,N_2), \cdots, s_i(1,1)) \cdots\cdots \\
&f_{s_i}(s_i(N_1,N_2)|s_i(N_1,N_2-1), \cdots, s_i(1,1))
\end{aligned} \tag{3}$$

This equation may be simplified by assuming a Markov model for the sources. We suppose hereafter that the sources are second-order Markov random fields, *i.e.* the conditional PDF of a pixel $s(n_1, n_2)$ given all the other pixels is equal to its conditional PDF given its 8 nearest neighbors (Fig. 2). From this assumption, it is clear that the conditional PDF of a pixel not situated on the boundaries, given all its predecessors (in the sense of sweeping trajectory) is equal to its conditional PDF given its three top neighbors and its left neighbor (squares in Fig. 2). In other words, if D_{n_1,n_2} is the set of pixel values $s_i(k,l)$ such that $\{k < n_1\}$ or $\{k = n_1, l < n_2\}$, then

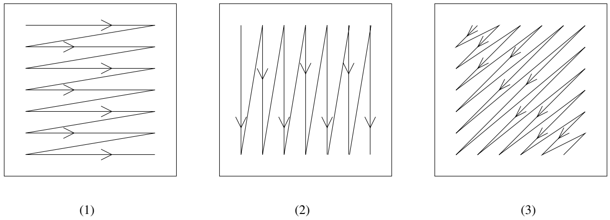

Fig. 1. Different sweeping possibilities

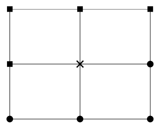

Fig. 2. Second-order Markov random field

$$f_{s_i}(s_i(n_1,n_2)|D_{n_1,n_2}) = f_{s_i}(s_i(n_1,n_2)|s_i(n_1,n_2-1), s_i(n_1-1,n_2+1), s_i(n_1-1,n_2), s_i(n_1-1,n_2-1)) \quad (4)$$

If N is sufficiently large, the conditional PDF of the pixels located on the left, top and right image boundaries (for which, the 4 mentioned neighbors are not available) may be neglected in (3). Supposing that the sources are stationary so that the conditional PDF (4) does not depend on n_1 and n_2, it follows from (4) that the decomposed joint PDF (3) can be rewritten as

$$f_{s_i}(s_i(1,1), s_i(1,2), \cdots, s_i(1,N_2), s_i(2,1), \cdots, s_i(N_1,N_2)) \simeq \prod_{n_1=2}^{N_1} \prod_{n_2=2}^{N_2-1} f_{s_i}(s_i(n_1,n_2)|s_i(n_1,n_2-1), s_i(n_1-1,n_2+1), s_i(n_1-1,n_2), s_i(n_1-1,n_2-1))$$

The log-likelihood function may be obtained by replacing the above PDF in (2) and taking the logarithm:

$$N\log(|\det(\mathbf{B})|) + \sum_{i=1}^{K} \sum_{n_1=2}^{N_1} \sum_{n_2=2}^{N_2-1} \log f_{s_i}(s_i(n_1,n_2)|s_i(n_1,n_2-1), s_i(n_1-1,n_2+1), s_i(n_1-1,n_2), s_i(n_1-1,n_2-1)) \quad (5)$$

Dividing the above cost function by N and defining the spatial average operator $E_N[.] = \frac{1}{N}\sum_{n_1=2}^{N_1}\sum_{n_2=2}^{N_2-1}[.]$, Equation (5) may be rewritten in the following simpler form

$$L_1 = \log(|\det(\mathbf{B})|) + E_N[\sum_{i=1}^{K} \log f_{s_i}(s_i(n_1,n_2)|s_i(n_1,n_2-1), s_i(n_1-1,n_2+1), s_i(n_1-1,n_2), s_i(n_1-1,n_2-1))]$$

In [2], the separating matrix $\mathbf{B}$ was obtained by maximizing the above cost function using a relative gradient ascent algorithm which is very time consuming. Here, we choose another approach which consists in solving the equation $\frac{\partial L_1}{\partial \mathbf{B}} = 0$ using a modified equivariant Newton-Raphson algorithm.

3 Estimating Equations and Their Solution

As shown in [2], the gradient of the cost function L_1 is equal to

$$\frac{\partial L_1}{\partial \mathbf{B}} = \mathbf{B}^{-T} - E_N[\sum_{(k,l)\in\Upsilon} \boldsymbol{\Psi}_{\mathbf{s}}^{(k,l)}(n_1,n_2).\mathbf{x}^T(n_1-k,n_2-l)] \tag{6}$$

where $\Upsilon = \{(0,0),(0,1),(1,-1),(1,0),(1,1)\}$ and the vector $\boldsymbol{\Psi}_{\mathbf{s}}^{(k,l)}(n_1,n_2)$ contains the conditional score functions of the K sources, which are denoted $\psi_{s_i}^{(k,l)}(n_1,n_2)$ hereafter for simplicity, and which read explicitly

$$\begin{aligned}
&\psi_{s_i}^{(k,l)}(n_1,n_2) = \psi_{s_i}^{(k,l)}(s_i(n_1,n_2)|s_i(n_1,n_2-1), s_i(n_1-1,n_2+1),\\
&s_i(n_1-1,n_2), s_i(n_1-1,n_2-1)) = \frac{-\partial}{\partial s_i(n_1-k,n_2-l)} \log f_{s_i}(s_i(n_1,n_2)|\\
&s_i(n_1,n_2-1), s_i(n_1-1,n_2+1), s_i(n_1-1,n_2), s_i(n_1-1,n_2-1))
\end{aligned} \tag{7}$$

Setting (6) to zero, then post-multiplying by $\mathbf{B}^T$ we obtain

$$E_N[\sum_{(k,l)\in\Upsilon} \boldsymbol{\Psi}_{\mathbf{s}}^{(k,l)}(n_1,n_2).\mathbf{s}^T(n_1-k,n_2-l)] = \mathbf{I} \tag{8}$$

This yields the $K(K-1)$ estimating equations

$$E_N[\sum_{(k,l)\in\Upsilon} \psi_{s_i}^{(k,l)}(n_1,n_2).s_j(n_1-k,n_2-l)] = 0 \qquad i \neq j = 1,\cdots,K \tag{9}$$

which determine $\mathbf{B}$ up to a diagonal and a permutation matrix. The other K equations $E_N[\sum_{(k,l)\in\Upsilon} \psi_{s_i}^{(k,l)}(n_1,n_2).s_i(n_1-k,n_2-l)] = 1 \quad i = 1,\cdots,K$ are not important and can be replaced by any other scaling convention.

The system of equations (9) may be solved using the Newton-Raphson algorithm. We propose a modified version of this algorithm which has the equivariance property, *i.e.* its performance does not depend on the mixing matrix.

To ensure the equivariance property, the adaptation gain must be proportional to the previous value of $\mathbf{B}$. Let $\tilde{\mathbf{B}}$ be an initial estimation of $\mathbf{B}$. We want to find a matrix $\boldsymbol{\Delta}$ so that the estimation $\hat{\mathbf{B}} = (\mathbf{I}+\boldsymbol{\Delta})\tilde{\mathbf{B}}$ be a solution of (9). To simplify the notations, we here only consider the case $K = 2$ but the same approach may be used for higher values of K. In the appendix, we show that the off-diagonal entries of $\boldsymbol{\Delta}$, δ_{12} and δ_{21}, are the solutions of the following linear system of equations

$$E_N[\sum_{(k,l)\in\Upsilon} \psi_{\hat{s}_1}^{(k,l)}(n_1,n_2).\tilde{s}_1(n_1-k,n_2-l)]\delta_{21}$$
$$+E_N[\sum_{(k,l)\in\Upsilon}\{\sum_{(i,j)\in\Upsilon}\frac{\partial\psi_{\hat{s}_1}^{(k,l)}(n_1,n_2)}{\partial s_1(n_1-i,n_2-j)}\tilde{s}_2(n_1-i,n_2-j)\}.\tilde{s}_2(n_1-k,n_2-l)]\delta_{12}$$
$$= -E_N[\sum_{(k,l)\in\Upsilon} \psi_{\hat{s}_1}^{(k,l)}(n_1,n_2).\tilde{s}_2(n_1-k,n_2-l)]$$

$$E_N[\sum_{(k,l)\in\Upsilon} \psi_{\hat{s}_2}^{(k,l)}(n_1,n_2).\tilde{s}_2(n_1-k,n_2-l)]\delta_{12}$$
$$+E_N[\sum_{(k,l)\in\Upsilon}\{\sum_{(i,j)\in\Upsilon}\frac{\partial\psi_{\hat{s}_2}^{(k,l)}(n_1,n_2)}{\partial s_2(n_1-i,n_2-j)}\tilde{s}_1(n_1-i,n_2-j)\}.\tilde{s}_1(n_1-k,n_2-l)]\delta_{21}$$
$$= -E_N[\sum_{(k,l)\in\Upsilon} \psi_{\hat{s}_2}^{(k,l)}(n_1,n_2).\tilde{s}_1(n_1-k,n_2-l)] \quad (10)$$

The computation of the coefficients δ_{12} and δ_{21} requires the estimation of the conditional score functions and their derivatives. In [2], we used a non-parametric method proposed in [7] involving the estimation of joint entropies using a discrete Riemann sum and third-order cardinal spline kernels. This estimator is very time and memory consuming and does not provide the derivatives of the score functions required for Newton-Raphson algorithm. In the following section, we propose another solution based on a third order polynomial parametric estimation of the score functions which is very fast and directly provides the derivatives of the score functions. Then, the terms δ_{12} and δ_{21} can be obtained by solving (10). The diagonal entries of $\boldsymbol{\Delta}$ are not important because they influence only the scale factors. Thus, we can fix them arbitrarily to zero.

4 Parametric Estimation of the Score Functions

Our parametric estimator of the conditional score functions is based on the following theorem, proved in [8] in the scalar case:

Theorem. If $\lim_{y_i\to\pm\infty} f_y(y_0,\cdots,y_q)g(y_0,\cdots,y_q) = 0$ [1] where f_y is the joint PDF of $y_0,\cdots .y_q$ and g is an arbitrary function of these variables, then

$$E[-\frac{\partial \log f_y(y_0,\cdots,y_q)}{\partial y_i}g(y_0,\cdots,y_q)] = E[\frac{\partial g(y_0,\cdots,y_q)}{\partial y_i}] \quad (11)$$

Following this theorem, if $g(y_0,\cdots,y_q,\mathbf{W})$ is a mean-square parametric estimator of the joint score function $\psi_{y_i}(y_0,\cdots,y_q) = -\frac{\partial \log f_y(y_0,\cdots,y_q)}{\partial y_i}$, its parameter vector $\mathbf{W}$, can be found from

[1] When $g(.)$ is bounded, this condition is satisfied for every real-world signal because its joint PDF tends to zero at infinity.

$$\mathbf{W} = argmin\{E[g^2(y_0, \cdots, y_q, \mathbf{W})] - 2E[\frac{\partial g(y_0, \cdots, y_q, \mathbf{W})}{\partial y_i}]\} \tag{12}$$

Note that the function to be minimized does not explicitly depend on the score function itself. In our problem, we want to estimate the conditional score functions. Each conditional score function can be written as the difference between two joint score functions:

$$\begin{aligned}\psi_{y_i}(y_0|y_1, \cdots, y_q) &= -\frac{\partial \log f_y(y_0, \cdots, y_q)}{\partial y_i} + \frac{\partial \log f_y(y_1, \cdots, y_q)}{\partial y_i} \\ &= \psi_{y_i}(y_0, \cdots, y_q) - \psi_{y_i}(y_1, \cdots, y_q)\end{aligned} \tag{13}$$

Each of two joint score functions in the above equation can be estimated using a parametric estimator which may be realized in different manners. In our work, we used the polynomial functions because of their linearity with respect to the parameters which simplifies the computations.

The conditional score functions used in our work being of dimension 5, they may be written as the difference between two joint score functions of dimensions 5 and 4 respectively. We used third-order polynomial functions for estimating them. The polynomial function modeling the 5-dimensional joint score function must contain all the possible terms in $\{1, (y_0, y_1, y_2, y_3, y_4), (y_0, y_1, y_2, y_3, y_4)^2, (y_0, y_1, y_2, y_3, y_4)^3\}$. Hence, it contains $\sum_{k=0}^{3} \binom{5+k-1}{k} = 56$ coefficients. In the same manner, the polynomial function modeling the 4-dimensional joint score function contains $\sum_{k=0}^{3} \binom{4+k-1}{k} = 35$ coefficients.

Our tests confirm that the above parametric estimator is much more faster, roughly 100 times, than the non-parametric estimator used in [2] and leads to the same performance.

5 Simulation Results

In the following experiments, we compare our method with two well-known algorithms: SOBI [6] and Pham-Garat [4]. SOBI is a second-order method which consists in jointly diagonalizing several covariance matrices evaluated at different lags. The Pham-Garat algorithm is based on a maximum likelihood approach which supposes that the sources are i.i.d. and therefore does not take into account their possible autocorrelation. For each experiment, the output Signal to Interference Ratio (in dB) was computed by $SIR = 0.5 \sum_{i=1}^{2} 10 \log_{10} \frac{E[s_i^2]}{E[(\hat{s}_i - s_i)^2]}$, after normalizing the estimated sources, $\hat{s}_i(n_1, n_2)$, so that they have the same variances and signs as the source signals, $s_i(n_1, n_2)$.

In the first experiment, we use artificial image sources of size 100×100 which satisfy exactly the considered Markov model. Two independent non-autocorrelated and uniformly distributed image noises, $e_1(n_1, n_2)$ and $e_2(n_1, n_2)$, are filtered by two autoregressive (AR) filters using the following formula:

$$\begin{aligned}s_i(n_1, n_2) = e_i(n_1, n_2) + \rho_{0,1} s_i(n_1, n_2 - 1) + \rho_{1,-1} s_i(n_1 - 1, n_2 + 1) \\ + \rho_{1,0} s_i(n_1 - 1, n_2) + \rho_{1,1} s_i(n_1 - 1, n_2 - 1)\end{aligned} \tag{14}$$

The coefficients $\rho_{i,j}$ are chosen to guarantee a sufficient stability condition. Thus, the coefficients of the first and the second filters are respectively fixed to $\{-0.5, 0.4, 0.5, 0.3\}$ and $\{-0.5, \rho_{1,-1}, 0.5, 0.3\}$. The coefficient $\rho_{1,-1}$ of the second filter may change in its stability interval, *i.e.* $[0.2, 0.6]$. Then, the source images $s_i(n_1, n_2)$ are mixed by the mixing matrix $\mathbf{A} = \begin{pmatrix} 1 & 0.99 \\ 0.99 & 1 \end{pmatrix}$. The mean of SIR over 100 Monte Carlo simulations as a function of the coefficient $\rho_{1,-1}$ of the second AR filter is shown in Fig. 3-a. Our algorithm outperforms the other two, whatever $\rho_{1,-1}$.

In the second experiment, the same non-autocorrelated and uniformly distributed image noises, $e_1(n_1, n_2)$ and $e_2(n_1, n_2)$, were generated and one of them was filtered by a symmetrical FIR filter. It is evident that the filtered signal is no longer a 2nd-order MRF. Then, the signals were mixed by the same matrix as in the first experiment. The mean of SIR as a function of the selectivity of the FIR filter is shown in Fig. 3-b. The performance of our method is always better than SOBI. It also outperforms Pham-Garat unless the filter selectivity is small so that the filtered signal is nearly uncorrelated. In the last experiment, the two real images of dimension 230×270 pixels, shown in Fig. 4, were mixed by the same matrix. It is clear that the working hypotheses are no longer true because the images are not stationary and cannot be described by 2nd-order MRFs. However, the images are autocorrelated and nearly cyclostationary because the correlation profiles on different circles are similar. Thus, the conditional score functions on different circles are nearly similar. Once more, the three mentioned algorithms were used for separating the sources. Our algorithm led to 57-dB SIR while SOBI and Pham-Garat led to 23-dB and 12-dB SIR respectively.

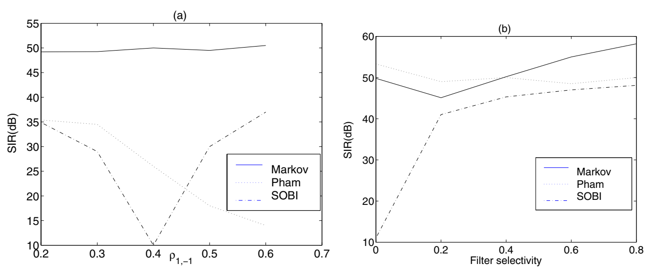

Fig. 3. Simulation results using (a) IIR and (b) FIR filters

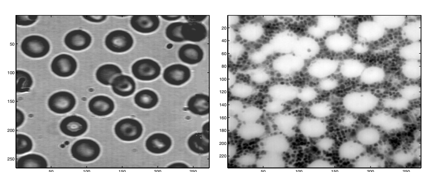

Fig. 4. Real-world images used in the experiment

6 Conclusion

In this paper, we made two major modifications in our markovian blind image separation algorithm *i.e.* utilization of a low-cost parametric score function estimator instead of the non-parametric estimator, and derivation of a modified equivariant Newton-Raphson algorithm for solving the estimating equations instead of maximizing the log-likelihood function by a relative gradient algorithm. These modifications led to a much faster algorithm and allowed us to perform more experiments using artificial and real-world data. These experiments confirmed the better performance of our method in comparison to the classical methods which ignore spatial autocorrelation or non-gaussianity of data.

References

1. S. Hosseini, C. Jutten and D.-T. Pham, Markovian source separation, *IEEE Trans. on Signal Processing*, vol. 51, pp. 3009-3019, 2003.
2. S. Hosseini, R. Guidara, Y. Deville and C. Jutten, Maximum likelihood separation of spatially autocorrelated images using a Markov model, in *Proc. of MAXENT'05*, San Jose, USA, August 2005.
3. E. E. Kuruoglu, A. Tonazzini and L. Bianchi, Source separation in noisy astrophysical images modelled by markov random fields, in *Proc. ICIP'04*, pp. 2701-2704.
4. D.-T. Pham and P. Garat, Blind separation of mixture of independent sources through a quasi-maximum likelihood approach, *IEEE Trans. on Signal Processing*, vol. 45, pp. 1712-1725, July 1997.
5. L. Tong, R. Liu, V. Soon and Y. Huang, Indeterminacy and identifiability of blind identification, *IEEE Trans. on Circuits Syst.*, vol. 38, pp. 499-509, May 1991.
6. A. Belouchrani, K. Abed Meraim, J.-F. Cardoso and E. Moulines, A blind source separation technique based on second order statistics, *IEEE Trans. on Signal Processing*, vol. 45, pp. 434-444, Feb. 1997.
7. D.-T. Pham, Fast algorithms for mutual information based independent component analysis, *IEEE Trans. on Signal Processing*, vol. 52, no. 10, Oct. 2004.
8. D.-T. Pham, P. Garat and C. Jutten, Separation of mixture of independent sources through a quasi-maximum likelihood approach, in *Proc. of EUSIPCO'92*, pp. 771-774, Brussels, August 1992.

Appendix. Derivation of Equations (10)

Post-multiplying $\hat{\mathbf{B}} = (\mathbf{I} + \boldsymbol{\Delta})\tilde{\mathbf{B}}$ by $\mathbf{x}$ we obtain $\hat{\mathbf{s}} = (\mathbf{I} + \boldsymbol{\Delta})\tilde{\mathbf{s}}$. Denoting $\boldsymbol{\Delta} = \begin{pmatrix} \delta_{11} & \delta_{12} \\ \delta_{21} & \delta_{22} \end{pmatrix}$, it implies that $\widehat{s}_1(n_1, n_2) = \widetilde{s}_1(n_1, n_2) + \delta_{11}\widetilde{s}_1(n_1, n_2) + \delta_{12}\widetilde{s}_2(n_1, n_2)$ and $\widehat{s}_2(n_1, n_2) = \widetilde{s}_2(n_1, n_2) + \delta_{21}\widetilde{s}_1(n_1, n_2) + \delta_{22}\widetilde{s}_2(n_1, n_2)$. Since $\hat{s}_1$ and $\hat{s}_2$ must satisfy the estimating equations (9), by replacing the above relations in the first estimating equation and considering (7) we obtain

$$E\Big[\sum_{(k,l)\in\Upsilon}\big\{\psi_{\widehat{s}_1}^{(k,l)}\big(\widetilde{s}_1(n_1,n_2)+\delta_{11}\widetilde{s}_1(n_1,n_2)+\delta_{12}\widetilde{s}_2(n_1,n_2)|\widetilde{s}_1(n_1,n_2-1)$$
$$+\delta_{11}\widetilde{s}_1(n_1,n_2-1)+\delta_{12}\widetilde{s}_2(n_1,n_2-1),\ldots,\widetilde{s}_1(n_1-1,n_2-1)$$
$$+\delta_{11}\widetilde{s}_1(n_1-1,n_2-1)+\delta_{12}\widetilde{s}_2(n_1-1,n_2-1)\big)\big\}.\big\{\widetilde{s}_2(n_1-k,n_2-l)$$
$$+\delta_{21}\widetilde{s}_1(n_1-k,n_2-l)+\delta_{22}\widetilde{s}_2(n_1-k,n_2-l)\big\}\Big]=0 \quad (15)$$

Using a first-order Taylor development of the score function, noting that the separated sources are independent at the vicinity of the solution, neglecting the terms containing the products of δ_{ij}, and neglecting δ_{22} with respect to 1, we obtain by some simple calculus the first equation in (10). The second equation can be derived by symmetry.