Unsupervised hidden state estimation and blind source separation using Auto-encoder RNN filter

*Note: Sub-titles are not captured in Xplore and should not be used

Clint Alex Steed
Department of Mechanical Engineering
Ulsan National Institute of Science and Technology
Ulsan, South Korea
https://orcid.org/0000-0001-7338-3696

Namhun Kim
Department of Mechanical Engineering
Ulsan National Institute of Science and Technology
Ulsan, South Korea
https://orcid.org/0000-0003-4429-2191

***Abstract*—**

This work proposes a deep learning estimator for unsupervised nonlinear hidden state estimation formulating the problem as blind source separation (BSS).

The model is composed of an auto encoder base d RNN to estimate the hidden state. The model is extended to the blind source separation using local losses to de-correlate the hidden signals. The problem is formulated such that the number of sources can be determined by varying the dimension of the hidden state signal. The solution is demonstrated on a number of simulations.

The simulation shows that the model is suitable for hidden state extraction. We find that the model will extract the hidden signals correctly when the correct dimensionality is selected , otherwise repeated hidden signals occur. Similarly, when applied to BSS, the model successfully separated multiple sources.

The model retains many of the limitations of BSS, such as being able to recover the component signals but not amplitude. The use of an auto-encoder limits the model to cases of over specified problems, where more sensors than hidden states are present, making it well suited for domains with multiple redundant sensors (drones, self-driving cars, etc.)

The filter provides much functionality by de-noising sensor signals, decoding sensor signals to either (1) a lower dimensional latent space performing (non-linear PCA) or (2) separates source signals (non-linear ICA) and forecasts predictions.

Keywords*—*Unsupervised learning, Filtering, Hidden State Estimation, Recurrent Neural Networks, Auto-Encoders, Blind source separation.

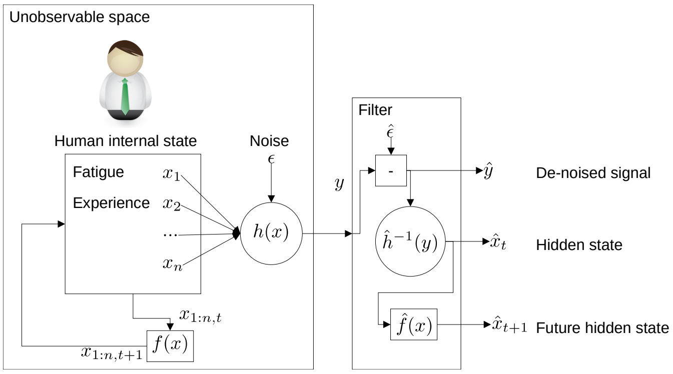
I. Introduction

Human centric systems are being actively researched as can be seen by a number of recent special issues [1]–[3], a EU report [4], and the raising cost of labor in manufacturing. This requirement, along with rapid change in systems encourages the use of automated control approaches. Yet these approaches are limited in their application of human control due to ethical issues.

As the issue stands "Systems Serve Humans" and not the other way around. Therefore it is important to have empathetic AI controlling systems. The authors believe this is possible by controlling systems based on human internal state variables like fatigue.

For example AI can be used to control experiments, ultimately reducing the number of experiments and in turn human effort. If the objective function was designed reduce cost effective it may be to the detriment of human operators and therefore unethical. The need to create a utility function that considers operator effort is limited by the ability to model the internal human states like fatigue.

This work is the starting point of investigating automatically extracting the human internal state. The main objective of the current investigation is to extract the number of independent sources. The problem is formulated as a blind source separation problem. A non-linear deep estimator is developed to extract the unobservable states.



II. Literature

A. Why are human unobservable states important

The issue of unobservable states is well known when measuring humans. Often a state like fatigue cannot be measured directly but the effects of them can be seen. A seminal study [5], [6] showed that the later in the day, the higher the risk of injury. The same study showed that consecutive work shifts also increased the risk of injury. This hints that there are multiple modalities to fatigue and we expect two separate sources, one for daily fatigue and another for weekly fatigue. This could be the case for other state variables.

A number of other works model human performance at the production level, using signals like throughput rate to model circadian rhythm [7], learning and forgetting [8], work-rest ratios [9] to name a few. Other work uses on person sensors like accelerometers, EMGs, and temperature sensors [10]. Yet more work uses biological samples from

XXX-X-XXXX-XXXX-X/XX/$XX.00 ©2023 IEEE

oral swabs to measure fatigue levels [11]. The issue is that many of these data acquisition methods are not feasible and it is not clear whether the information provided overlaps. The hope is that using deep learning to automatically learn the mapping between sensors and internal state may provide some insight into selecting a practical combination of sensors for a specific application. To do so BSS must be extended to consider dynamic systems where signals vary of time and hidden state variable interact.

B. *Blind source separation*

The blind source separation problem is sometimes better described by the cocktail party problem. Imagine numerous people talking, resulting in the recipient receiving a mixed sound signal and having to discern between different conversations. The authors user this term as a problem formulation rather than a collection of methods. BSS methods have been used for audio source separation [12] and signal processing [13].

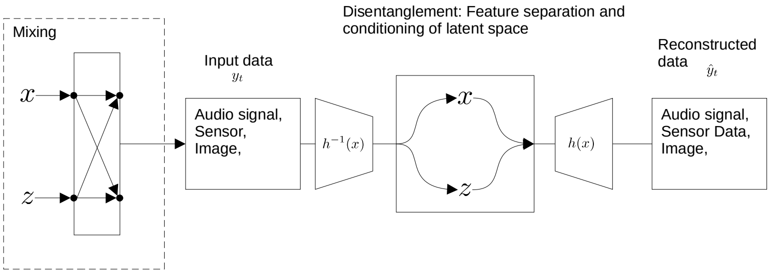
BSS is often an ill conditioned problem, resulting in numerous solution. Further specifying constraints has the potential to reduce this. Some known issues are that although the signal can be recovered, the amplitude cannot. This is overcome by a number a local losses discussed shortly. Another known limitation is the assumption that at most one signal is Gaussian.

One technique for achieving BSS is Independent Component Analysis (ICA), an extension to the well known Principal Component Analysis (PCA).

C. *Deep blind source separation as high level feature separation*

The authors argue that with deep learning, a number of high-level tasks can be represent as a BSS problem, image processing and generation techniques being the easiest to visualize. These works provide insights into methods of conditioning the latent space, as tasks often involve modifying high level features to generate new images.

For example, [14] explicitly separated facial identities from emotions and was able to reconstruct faces with different emotions. [15] separated blur, noise, and compression image distortions. One of the more interesting of these is Fader networks [16] which allows a sliding attributes to adjust the intensity of a feature (e.g. young→ old).


Mixing
x
z
Disentanglement: Feature separation and conditioning of latent space
Input data
y_t
Audio signal, Sensor, Image,
$h^{-1}(x)$
x
z
$h(x)$
Reconstructed data
$\hat{y}_t$
Audio signal, Sensor Data, Image,

Disentanglement is used to describe the linearity of separation of features, styles, and other information in the latent space [17], [18]. This disentanglement resembles source separation. A number of methods for estimating disentanglement are suggested, [19] suggests developing a linear classifier for this very purpose, [20] proposes cluster separability, [21] uses a probabilistic total correlation penalty and therefore requires sampling, [16] use a discriminator to much success.

Although auto-encoders are a commonly used for this problem [17], [22], [23], some models do not employ AE [18], making disentanglement comparison difficult.

D. *Hebbian learning inspired local losses*

One neural learning algorithm which has shown lots of promise in this area is Hebb learning. Although it is not used in this work due to back-propagation tools being more mature. The insights found in Hebb learning motivate the choices for local losses here.

Hebbian learning is best described by the adage “Neurons that fire together, wire together” [24] . The Hebbian learning interpretation of this strengthens of pre-synaptic and post-synaptic pairs that fire together. This results in learning the principal components [25]. On the other hand, Anti-Hebbian learning weakens pre-synaptic and post-synaptic pairs that don’t fire together, resulting in de-correlation which can be used for BSS [26], [27]. This Anti-Hebbian learning can be imitated using an auto-encoder with the inappropriately named Decov loss [14], [28].

Hebbian learning has also addressed some of the other limitations in BSS, by conditioning the source signals. Unscaled source amplitude is addressed by enforcing unit variance [27], this in turn inspires the use of unit variance local loss use here. Similarly, zero mean source signal is typically achieved by whitening the data, instead we use a small zero-mean loss.

E. *Auto-encoder implications on sensor design*

The auto-encoder is selected as the starting point for the model because there is strong evidence that it performs non-linear principal component analysis (PCA) [29]. The intuition here is to use a de-correlation to move toward non-linear Independent component analysis (ICA), one of the better known methods for BSS.

The auto-encoder does however place some restrictions on our sensor selection. It is assumed that the number of sensor signals is greater than the number of source signals, $m > n$ where $x \in R^n$ $y \in R^m$. This is not unreasonable, as the auto-encoder also performs de-noising. Typically in estimation, several noisy sensors are preferred to a few high quality ones due to cost and redundancy.

F. *Deep temporal estimators*

Most BSS work consider static solutions. For example Fourier transform and have the limitation on time varying signals and latent state interaction. Formulating this problem as a dynamical system has the potential to relax these two limitation.

Well known filters like the Kalman filter and extended Kalman filter have been widely applied, but their linear limitations are known [30], [31]. Another generation of filters use computationally intensive monte carlo simulations to estimate non-linear behavior [32]. Deep filtering techniques tend to spend this computational cost upfront by learning filtering parameters and estimating functions, resulting in cost effective inference. Here either the functions or parameters are learned. A desirable trait with deep filters its the ability to include prior known information , usually in the form of partially known dynamics [33].

III. Theory

The figure that follows depicts the decisions made when developing the mode. Starting from a standard auto-encoder, moving towards a supervised temporal estimator, and then an unsupervised estimator.

We begin by developing the model, then describe the local losses required to shape the latent state.

A. Filtering model

The model is developed starting from a standard auto-encoder with the reconstruction loss. Some evidence indicates this performs non-linear PCA [].

$$L_1 = ||y - \hat{y}||$$

Next we create a supervised filter by adding an evolution/transition function $f(x)$ and loss $L_2 = ||\hat{x}_{t+1} - x_{t+1}||$. Some important notes here is that (1) we assume our sensor dimension to be higher than our latent dimension $y \in R^m, x \in R^n, m > n$. This is advantages since it is common to have numerous redundant sensors for noise reduction and reliability. This model assumes that the unobservable state data x is available for training, but in the next step we lift this condition.

$$L_2 = ||\hat{h}^{-1}(y_t) - x_{t+1}||$$

The final step is unrolling in a similar way to other recurrent neural networks. Here a loss penalizes the error between sequential predictions of the model, specifically the encoded temporal-prediction from the current time $\hat{x}_{t+1}|\hat{x}_t$ and next encoded prediction $\hat{x}_{t+1}|y_{t+1}$. This removes the requirement for the hidden state data X, relying only on Y returning to an unsupervised learning problem. The cost of this is that batches of at-least 2 sequential data-points be used.

$$L_2' = ||\hat{f}(\hat{h}^{-1}(y_t)) - \hat{h}^{-1}(y_{t+1})||$$

If the equation $f(x)$ is known it can be substituted for the neural network.

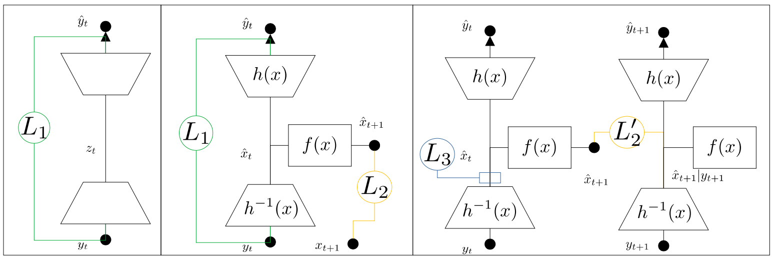


Figure x: From left (1) shows a standard auto-encoder, (2) the supervised filter, and (3) the unsupervised filter.

Since we do not supply $f(x)$, the model must infer a solution. This is a poorly conditioned problem so numerous solutions exist. We make not reservations about the form of $f(x)$ and x. In fact in many cases it is impossible to recover the amplitude of signals in x. The interesting question raised here is, “what happens when we vary n?”

B. Local losses

As mentioned before the amplitude for the signals often cannot be recovered. The widely accepted strategy in the community is to calibrate the source signals to some domain, for example $x \in [0, 1]$. In this work a number of losses are used. These losses are local to the mini-batch used in training but we have found it to be sufficient in our testing.

Firstly a mean loss encourages zero mean $L_\mu(x) = |x|$. This loss is typically an order of magnitude smaller than other losses.

The second loss ensures unit variance $L_\sigma(x) = |1 - \sigma(x)|$.

Finally a loss do-correlates x signals.
$L_{DeCov}(x) = \frac{1}{2}(||C|| - ||diag(C)||)$, where
$C_{i,j} = \frac{1}{N}(x_i, \mu i)(x_j, \mu_j)$. The intuition for this choice is moving from PCA to ICA. The resulting local losses can then be weighted and summed,
$L_3(x) = w_1 L_\mu + w_2 L_\sigma + w_3 L_{Decov}$.

C. The dimensionality of the latent space

Given this filter the challenge is now to choose the dimensionality of x, n where $x \in R^n$. The dimensionality of y, m where $y \in R^m$, is dictated by the sensors. We will arbitrarily select n such that we can learn more about the system. Note that we assume knowing nothing about the system, we only have the observable data Y. The filter then should infer all other components.

IV. Methodology

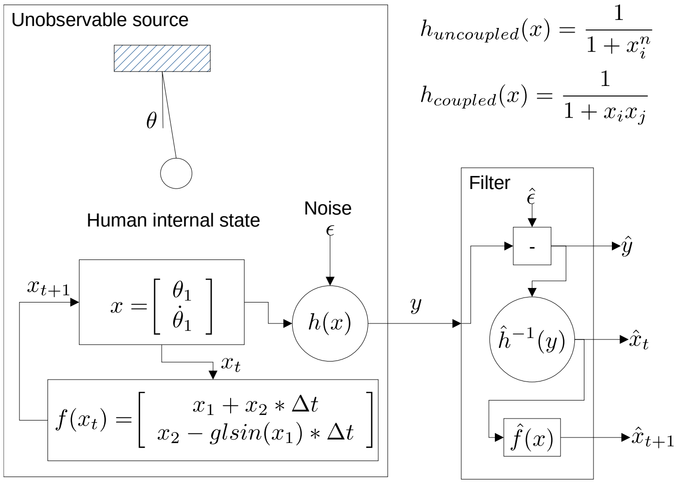
In order to validate the models performance two separate simulations are conducted. The first attempts to answer “Can the model extract multiple signals from a single source?” The second simulations extends to model to extract the signals of multiple sources. The authors suspect that the model cannot do both of these due to conflicting losses.

A. Model

In order to evaluate the filters behavior it is tested on a toy problem of one pendulum acting as a single source. The state is generated by the system transition $x_{t+1} = f(x_t)$. The model receives the sensor signal {y} which is mixed and noise added according to $y = h(x)$. The goal of the model is then to estimate the transition function $\hat{f}(x)$ and the state estimation function $\hat{x} = h^{-\hat{1}}(y)$.

Two mixing strategies are considered. Firstly, independent non-linear mixing via $h(x) = \frac{1}{1+|x_i|}$, which tests the models ability to perform non-linear estimation. Next, a non-linear combination mixing $h(x) = \frac{1}{1+|x_i x_j|}$, testing source separation. These sensor models change the signal significantly and do not allow negative values in these simulations. This has an impact on the resulting transition function.

We explore the selection of latent space and encounter the repeated signals issue. A more verbose figure is presented in the appendix.


Unobservable source
θ
Human internal state
Noise
ϵ
x_{t+1}
$x = \begin{bmatrix} \theta_1 \\ \dot{\theta}_1 \end{bmatrix}$
$h(x)$
y
x_t
$f(x_t) = \begin{bmatrix} x_1 + x_2 * \Delta t \\ x_2 - glsin(x_1) * \Delta t \end{bmatrix}$
$h_{uncoupled}(x) = \frac{1}{1 + x_i^n}$
$h_{coupled}(x) = \frac{1}{1 + x_i x_j}$
Filter
$\hat{\epsilon}$
-
$\hat{y}$
$\hat{h}^{-1}(y)$
$\hat{x}_t$
$\hat{f}(x)$
$\hat{x}_{t+1}$

B. *Multiple sources*

Next a system consisting of two pendulums moving at different frequencies is used. This section tests source separation.

A number of systems are used. Firstly the pendulum is selected for its familiarity. Also the Van der pol attractor is selected as it can be tuned to represent non-symmetric waves [34]. Finally, the a triangular wave it used due to its discontinuous nature.

V. Results

A. *Single source pendulum state estimation*

As expected the model infers principal signals. The leading and lagging relationship between the position and velocity was learned. We also see that noise is present in the result, it is not clear if this can be remedied with common techniques like regularization. The relative increasing magnitude is also captured, showing time varying signals are captured.

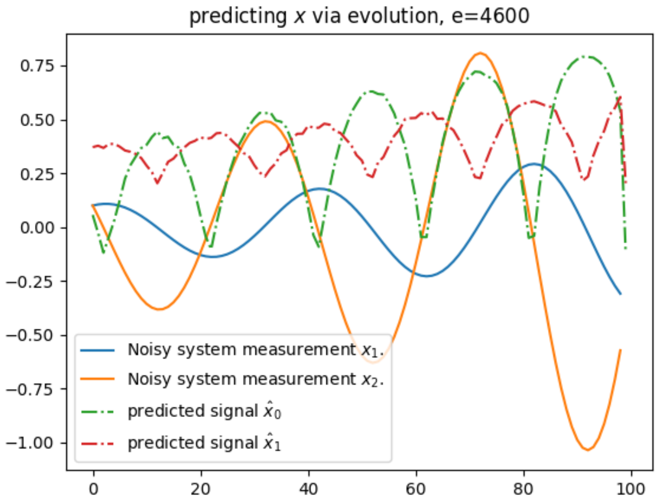


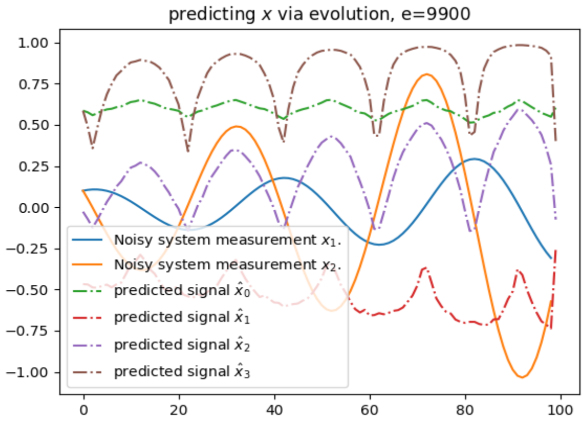
Figure x: showing the inference of the two hidden states (position and velocity) of the pendulum. The mean

This result would indicate that the model is sufficient for decoding and predicting some indicators of the hidden state.

B. *Varying the latent space dimensionality*

It was noticed that when n=1, the result was unique, that is “for different runs with random initialization, the resulting signal x was the same”. However for n>1, the results were not unique, the mean and sign of the signals would vary.

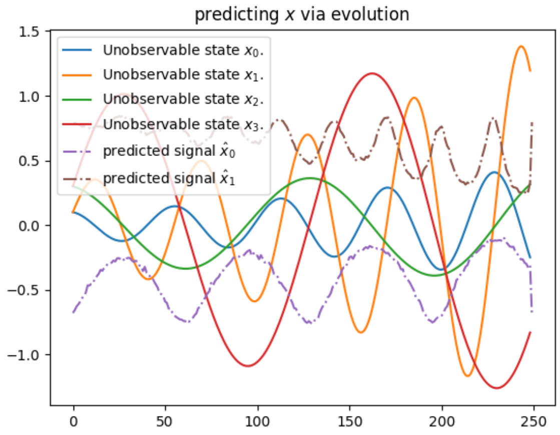
When increasing n>2 repeated signals occurred. This is likely due to repeated principals components.



Identifying unique signals can be used to select the appropriate principal dimension size n. Currently this is done through visual inspection, this is not ideal.

C. *Source separation*

As mentioned before the process is repeated with multiple sources and de-correlation added to the model to determine whether the model can perform blind source separation.

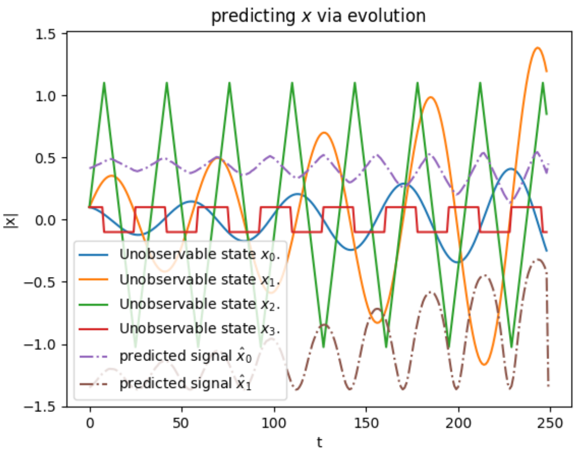


The figure above clearly shows that the separated signals are observed. It is also clear that signals are not immune to noise. Again the amplitudes are not repeatable between runs.

D. *Common systems*

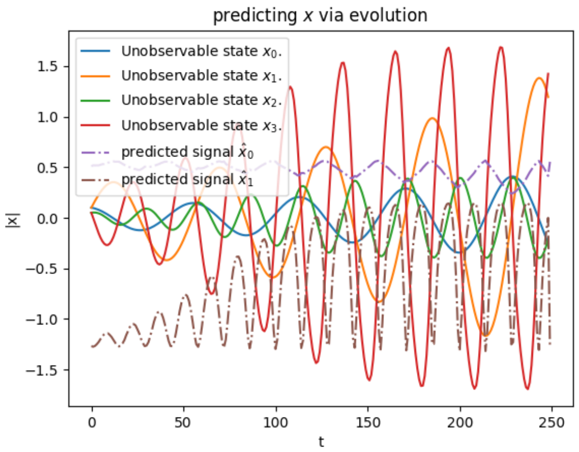
The triangular wave was also reproduced, showing the model can learn signals that are not smooth.

A Van der Pol attractor was used and the model was able to reconstruct these signals. Showing it can model non-symmetric waves and limit cycles.



E. Conclusion and further work

The model proposed here combines a dynamical system and blind source separation to enable automated decoding and hidden state estimation using unsupervised learning. This has numerous merits. The model can incorporate a transition function if one is known, but can learn one if it not available. The learned transition function amplitudes may not be recoverable and is influenced by the sensor selection. This posses issues if we hope to control based on these relative hidden state values.



The selection of the hidden state dimensionality was used to determine the number of independent sources. Visual inspection was used to select the dimensionality, but this is not ideal and a better means selection should be investigated.

References

[1] N. A. Stanton, “Special issue on human factors and ergonomics methods,” *Hum. Factors Ergon. Manuf.*, vol. 32, no. 1, pp. 3–5, 2022, doi: 10.1002/hfm.20943.

[2] F. Sgarbossa, E. Grosse, W. P. Neumann, and C. Berlin, “Call for Papers: Human-centric production and logistics systems,” *Int. J. Prod. Res.*, 2022, [Online]. Available: https://www.callforpapers.co.uk/human-factors-i50

[3] Wang Baicun, Peng Tao, Xi Vincent Wang, Thorsten Wuest, David Romero, and Lihui Wang, Eds., “Human-centric Smart Manufacturing: Trends, Issues and Challenges,” *J. Manuf. Syst.*, 2021.

[4] “Industry 5.0: Towards more sustainable, resilient and human-centric industry.” https://research-and-innovation.ec.europa.eu/news/all-research-and-innovation-news/industry-50-towards-more-sustainable-resilient-and-human-centric-industry-2021-01-07_en (accessed Sep. 20, 2022).

[5] S. Folkard and D. A. Lombardi, “Modeling the impact of the components of long work hours on injuries and ‘accidents,’” *Am. J. Ind. Med.*, vol. 49, no. 11, pp. 953–963, 2006, doi: 10.1002/ajim.20307.

[6] D. Fischer, D. A. Lombardi, S. Folkard, J. Willetts, and D. C. Christiani, “Updating the ‘Risk Index’: A systematic review and meta-analysis of occupational injuries and work schedule characteristics,” *Chronobiol. Int.*, vol. 34, no. 10, pp. 1423–1438, 2017, doi: 10.1080/07420528.2017.1367305.

[7] T. Åkkerstedt, S. Folkard, and C. Portin, “Predictions from the Three-Process Model of Alertness,” *Aviat. Space Environ. Med.*, vol. 75, no. 3, 2004.

[8] M. Y. Jaber, Z. S. Givi, and W. P. Neumann, “Incorporating human fatigue and recovery into the learning–forgetting process,” *Appl. Math. Model.*, vol. 37, no. 12–13, pp. 7287–7299, Jul. 2013, doi: 10.1016/j.apm.2013.02.028.

[9] F. Fruggiero, S. Riemma, Y. Ouazene, R. Macchiaroli, and V. Guglielmi, “Incorporating the Human Factor within Manufacturing Dynamics,” *IFAC-Pap.*, vol. 49, no. 12, pp. 1691–1696, 2016, doi: 10.1016/j.ifacol.2016.07.825.

[10] Z. Sedighi Maman, M. A. Alamdar Yazdi, L. A. Cavuoto, and F. M. Megahed, “A data-driven approach to modeling physical fatigue in the workplace using wearable sensors,” *Appl. Ergon.*, vol. 65, pp. 515–529, 2017, doi: 10.1016/j.apergo.2017.02.001.

[11] E. Bal, O. Arslan, and L. Tavacioglu, “Prioritization of the causal factors of fatigue in seafarers and measurement of fatigue with the application of the Lactate Test,” *Saf. Sci.*, vol. 72, pp. 46–54, 2015.

[12] M. Pal, R. Roy, J. Basu, and M. S. Bepari, “Blind source separation: A review and analysis,” *2013 Int. Conf. Orient. COCOSDA Held Jointly 2013 Conf. Asian Spok. Lang. Res. Eval. O-COCOSDACASLRE 2013*, 2013, doi: 10.1109/ICSDA.2013.6709849.

[13] J. He, W. Chen, and Y. Song, “Single Channel Blind Source Separation Under Deep Recurrent Neural Network,” *Wirel. Pers. Commun.*, vol. 115, no. 2, pp. 1277–1289, Nov. 2020, doi: 10.1007/S11277-020-07624-4/FIGURES/6.

[14] M. Cogswell, F. Ahmed, R. Girshick, L. Zitnick, and D. Batra, “Reducing overfitting in deep networks by decorrelating representations”.

[15] S. Bianco, L. Celona, and P. Napoletano, “Disentangling Image distortions in deep feature space,” *Pattern Recognit. Lett.*, vol. 148, pp. 128–135, Aug. 2021, doi: 10.1016/J.PATREC.2021.05.008.

[16] G. Lample, N. Zeghidour, N. Usunier, A. Bordes, L. Denoyer, and M. ’ A. Ranzato, “Fader Networks: Manipulating Images by Sliding Attributes,” in *Advances in Neural Information Processing Systems 30 (NIPS)*, 2017. Accessed: Dec. 27, 2022. [Online]. Available: https://github.com/facebookresearch/FaderNetworks

[17] Y. Liu, M. De Nadai, J. Yao, N. Sebe, B. Lepri, and X. Alameda-Pineda, “GMM-UNIT: Unsupervised Multi-Domain and Multi-Modal Image-to-Image Translation via Attribute Gaussian Mixture Modeling”.

[18] T. Karras NVIDIA and S. Laine NVIDIA, “#StyleGAN - A Style-Based Generator Architecture for Generative Adversarial Networks Timo Aila NVIDIA,” *Cvpr 2019*, 2019, [Online]. Available: https://github.com/NVlabs/stylegan

[19] I. Higgins *et al.*, “β-VAE: LEARNING BASIC VISUAL CONCEPTS WITH A CONSTRAINED VARIATIONAL FRAMEWORK,” in *International conference on learning representations, ICLR*, 2017. Accessed: Dec. 28, 2022. [Online]. Available: https://openreview.net/forum?id=Sy2fzU9gl

[20] B. Liu, Y. Zhu, Z. Fu, G. De Melo, and A. Elgammal, “Disentangling GAN with One-Hot Sampling and Orthogonal Regularization”, Accessed: Dec. 27, 2022. [Online]. Available: www.aaai.org

[21] H. Kim and A. Mnih, “Disentangling by Factorising,” in *NIPS, Learning Disentangled Representations: From Perception to Control Workshop*, 2017.

[22] Y. F. Zhou, R. H. Jiang, X. Wu, J. Y. He, S. Weng, and Q. Peng, “BranchGAN: Unsupervised Mutual Image-to-Image Transfer with A Single Encoder and Dual Decoders,” *IEEE Trans. Multimed.*, vol. 21, no. 12, pp. 3136–3149, Dec. 2019, doi: 10.1109/TMM.2019.2920613.

[23] M. Y. Liu, T. Breuel, and J. Kautz, “Unsupervised image-to-image translation networks,” *Adv. Neural Inf. Process. Syst.*, vol. 2017-Decem, no. Nips, pp. 701–709, 2017.

[24] R. G. M. Morris, “D.O. Hebb: The Organization of Behavior, Wiley: New York; 1949,” *Brain Res. Bull.*, vol. 50, no. 5–6, p. 437, Nov. 1999, doi: 10.1016/S0361-9230(99)00182-3.

[25] E. Oja, “Simplified neuron model as a principal component analyzer,” *J. Math. Biol.*, vol. 15, no. 3, pp. 267–273, Nov. 1982, doi: 10.1007/BF00275687.

[26] A. Carlson, “Biological Cybernetics Anti-Hebbian learning in a non-linear neural network,” 1990.

[27] C. Pehlevan, S. Mohan, and D. B. Chklovskii, “Blind Nonnegative Source Separation Using Biological Neural Networks,” *Neural Comput.*, vol. 29, no. 11, pp. 2925–2954, Nov. 2017, doi: 10.1162/neco_a_01007.

[28] B. Cheung, J. A. Livezey, A. K. Bansal, and B. A. Olshausen, “Discovering Hidden Factors of Variation in Deep Networks”.

[29] G. Alain, Y. Bengio, A. Courville, R. Fergus, and C. Manning, “What Regularized Auto-Encoders Learn from the Data-Generating Distribution,” *J. Mach. Learn. Res.*, vol. 15, pp. 3743–3773, 2014.

[30] R. E. Kalman, “A New Approach to Linear Filtering and Prediction Problems,” *J. Basic Eng.*, vol. 82, no. 1, p. 35, 1960, doi: 10.1115/1.3662552.

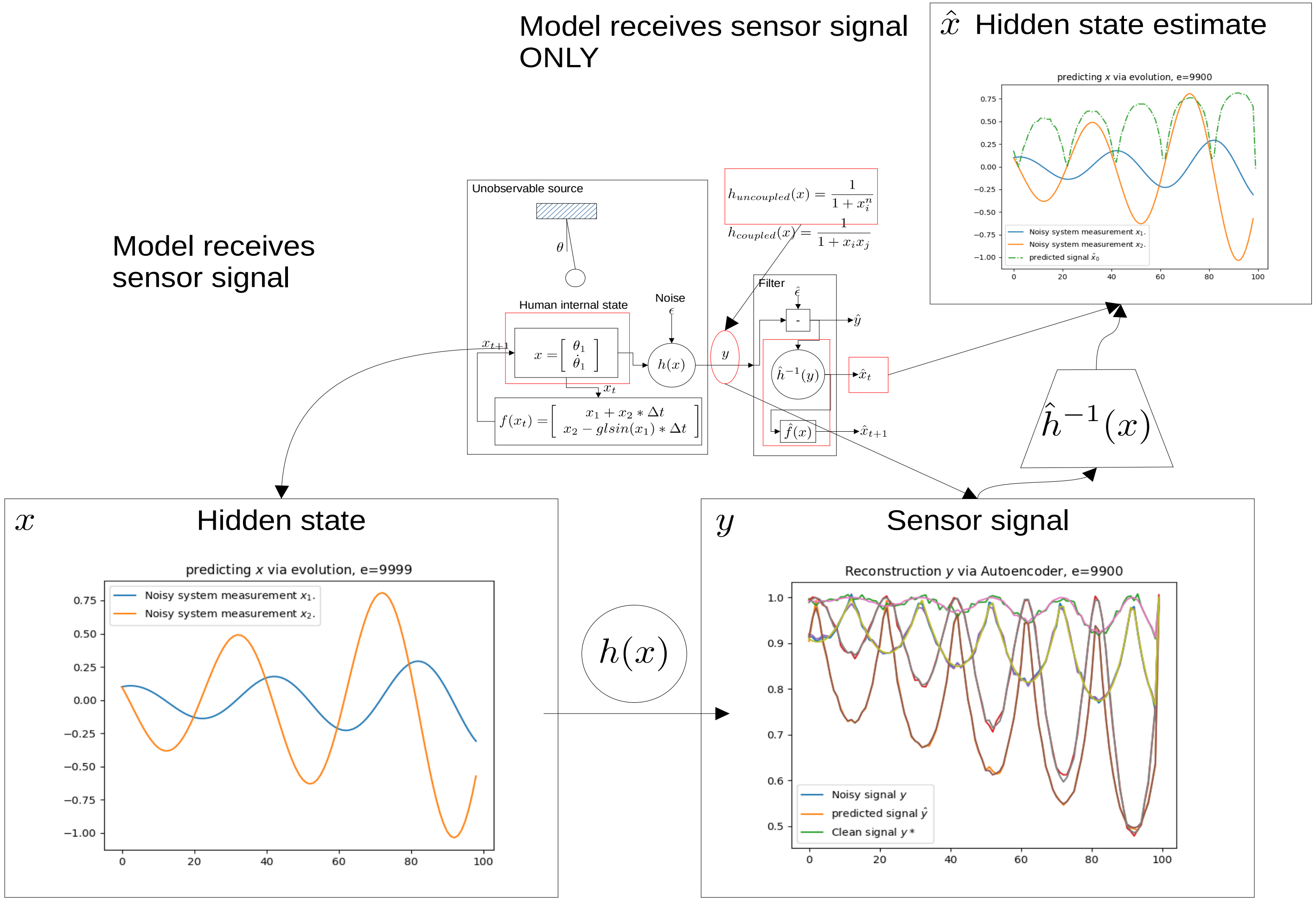
[31] B. A. McElhoe, “An assessment of the navigation and course corrections for a manned flyby of mars or venus,” *IEEE Trans. Aerosp. Electron. Syst.*, vol. AES-2, no. 4, pp. 613–623, 1966, doi: 10.1109/TAES.1966.4501892.

[32] P Del Moral, “Nonlinear Filtering: Interacting Particle Resolution,” *Markov Process. Relat. Fields*, vol. 2, no. 4, pp. 555–580, 1996.

[33] G. Revach, N. Shlezinger, X. Ni, A. L. Escoriza, R. J. G. van Sloun, and Y. C. Eldar, “KalmanNet: Neural Network Aided Kalman Filtering for Partially Known Dynamics,” *IEEE Trans. Signal Process.*, vol. 70, pp. 1532–1547, 2022, doi: 10.1109/TSP.2022.3158588.

[34] K. Hassan, *Nonlinear Systems*. Prentice-Hall, 2002.

Appendix A:
Experimental procedure



Model receives sensor signal ONLY

Model receives sensor signal

$\hat{x}$ Hidden state estimate

x Hidden state

y Sensor signal

$h(x)$

$\hat{h}^{-1}(x)$

predicting x via evolution, e=9900
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
0
20
40
60
80
100
Noisy system measurement x_1.
Noisy system measurement x_2.
predicted signal $\hat{x}_0$
Unobservable source
θ
Human internal state
Noise
ϵ
x_{t+1}
$x = \begin{bmatrix} \theta_1 \\ \dot{\theta}_1 \end{bmatrix}$
$h(x)$
x_t
$f(x_t) = \begin{bmatrix} x_1 + x_2 * \Delta t \\ x_2 - glsin(x_1) * \Delta t \end{bmatrix}$
$h_{uncoupled}(x) = \frac{1}{1 + x_i^n}$
$h_{coupled}(x) = \frac{1}{1 + x_i x_j}$
y
Filter
$\hat{\epsilon}$
-
$\hat{y}$
$\hat{h}^{-1}(y)$
$\hat{x}_t$
$\hat{f}(x)$
$\hat{x}_{t+1}$
predicting x via evolution, e=9999
Noisy system measurement x_1.
Noisy system measurement x_2.
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
0
20
40
60
80
100
Reconstruction y via Autoencoder, e=9900
1.0
0.9
0.8
0.7
0.6
0.5
Noisy signal y
predicted signal $\hat{y}$
Clean signal $y*$
0
20
40
60
80
100