

**Detecting the phase transition in a strongly interacting Fermi gas by unsupervised machine learning**D. Eberz <sup>1</sup>, M. Link <sup>1</sup>, A. Kell <sup>1</sup>, M. Breyer <sup>1</sup>, K. Gao <sup>1,2</sup> and M. Köhl <sup>1</sup><sup>1</sup>*Physikalisches Institut, University of Bonn, Wegelerstraße 8, 53115 Bonn, Germany*<sup>2</sup>*Department of Physics, Renmin University of China, Beijing 100872, China*

(Received 11 April 2023; accepted 20 October 2023; published 4 December 2023)

We study the critical temperature of the superfluid phase transition of strongly interacting fermions in the crossover regime between a Bardeen-Cooper-Schrieffer superconductor and a Bose-Einstein condensate of dimers. To this end, we employ the technique of unsupervised machine learning using an autoencoder neural network, which we directly apply to time-of-flight images of the fermions. We extract the critical temperature of the phase transition from trend changes in the data distribution revealed in the latent space of the autoencoder bottleneck.

DOI: [10.1103/PhysRevA.108.063303](https://doi.org/10.1103/PhysRevA.108.063303)

An ensemble of attractively interacting fermions exhibits a phase transition to a superfluid state below a critical temperature  $T_C$ . The exact temperature at which the phase transition occurs depends on the microscopic details, such as interparticle interactions and correlations. For weak attractive interactions, in the Bardeen-Cooper-Schrieffer (BCS) regime, the phase transition is governed by the opening of a gap due to Cooper instability near the Fermi level. The critical temperature in this regime decays exponentially with decreasing interaction strength. If the system supports a dimer bound state between two fermions, the ensemble can form a molecular Bose-Einstein condensate (BEC). The critical temperature of this state converges towards the value of a weakly repulsive BEC for decreasing interaction strength. These two regimes are known as the limits of the BEC-BCS crossover, connected by the unitarity regime around the point of diverging scattering length. In the regime of strong interaction strength around unitarity, the determination of the critical temperature is a field of ongoing research [1–9].

Detecting the phase transition over a wide range of interactions has been difficult. Only on the BEC side of the crossover does the conventional technique of detecting the bimodal momentum distribution of the dimers directly reveal the condensate [10–12]. In contrast, at unitarity a measurement of the equation of state and thermodynamic quantities unveiled the critical temperature [13]. On the BCS side of the crossover, the Cooper pairs break upon release from the trap and, therefore, the so-called rapid-ramp technique has been developed to convert these pairs to tightly bound dimers [14–17]. Whether or not the rapid-ramp technique closely reflects the situation of the trapped gas depends crucially on the adiabaticity of the ramp, and an accurate verification of this is very difficult. A direct detection of the superfluid signature in the momentum distribution of the fermions, on the other hand, is obscured by finite temperature, collisions during ballistic expansion, and the shape of the trapping potential. Recently, we have demonstrated that using supervised learning of deep neural networks, the condensate fraction and hence the critical temperature over a wide range of interactions can be detected directly from the momentum distribution of the fermions [18].

However, this method still relies on the rapid-ramp technique for labeling the training data.

In this work, we measure the critical temperature of the superfluid phase transition by employing unsupervised machine learning directly on time-of-flight images. Unsupervised machine learning is a technique that does not require labeling the data during training of the network and hence is unbiased. This is a significant advantage because the generation of labels is a potential source of error. To this end, we employ a deep neural network as an autoencoder, as illustrated in Fig. 1. The autoencoder comprises an encoder and decoder network as well as a bottleneck layer in the middle, with the input of the encoder being of the same dimension and shape as the output of the decoder. The bottleneck layer connects the input and output layers and, preferably, has much lower dimensionality than both input and output. Further, the encoder neural network is trained to compress the features of the input data (e.g., a picture) to the few neurons of the bottleneck, while the decoder neural network uses the bottleneck as an input to replicate the original input signal as accurately as possible. In this approach, a low-dimensional bottleneck will enforce an efficient representation of the data into a few meaningful parameters. If the autoencoder is successful, the latent space of the neuron outputs in the bottleneck contains relevant information, which can be interpreted to classify the input data. This is especially noteworthy because, by construction, the autoencoder does not take any additional information other than the raw data into account. So far, unsupervised machine learning has been used on theoretical data with learning by “confusion” [19], principal component analysis [19–21], autoencoders [21,22], and for the classification of topological phases in experimental data [23].

Experimentally, we prepare a quantum gas of  $\sim 3 \times 10^5$  atoms per spin state in the two lowest hyperfine states |1) and |2) of  $^6\text{Li}$  in a crossed-beam optical dipole trap [24]. We adjust the interaction strength of the sample by utilizing a magnetically controlled Feshbach resonance, and the temperature by controlled heating from a time-dependent variation of the trapping potential [18].

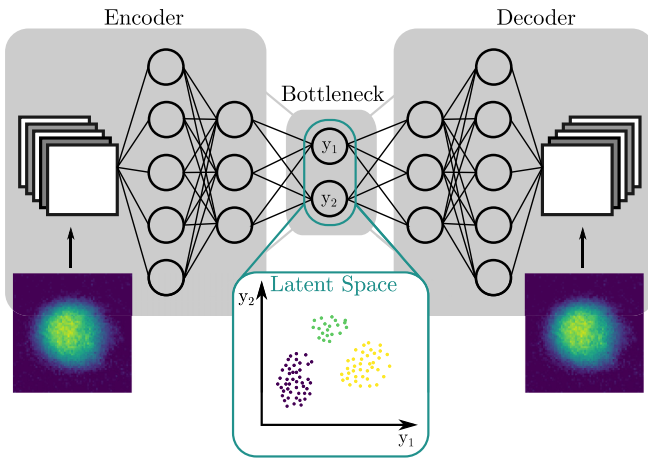


FIG. 1. Architecture and training of an autoencoder network. By keeping the number of neurons in the bottleneck low and training the network to reproduce its input, a low-dimensional representation of the input data can be generated. The output of the bottleneck can then be accessed to search for features in the data structure.

This approach enables us to tune the interaction strength and temperature independently of each other. In addition, we measure the central density and temperature of the cloud by performing an inverse Abel transform to the column density of *in situ* images. After reconstructing the three-dimensional (3D) density distribution in the trap, we use the central density to calculate the homogeneous Fermi energy and fit a virial expansion of the equation of state to the edge of the cloud to determine the temperature. The thermalized cloud is detected by absorption imaging after 5 ms of time-of-flight, which serves as the raw data for the autoencoder analysis. In the analysis, the calibration is used to assign time-of-flight images of variable heating time with a corresponding Fermi energy and temperature. Further details regarding the detection and calibration of the data are given in our previous work, which shares the same data set [18].

In total, we accumulate 5031 time-of-flight pictures at different interaction strengths and temperatures (see Fig. 1), which we split into 90% for training and 10% for validation after random shuffling. The network is written within the TensorFlow framework [25] and trained with stochastic gradient descent via the Adam optimizer [26]. The architecture of the autoencoder resembles two mirrored convolutional neural networks connected by a two-dimensional bottleneck layer. Both networks have independent weights and are trained simultaneously in the form of the autoencoder to reconstruct the time-of-flight images. Later, the outputs of the bottleneck are extracted to classify the data. The details of the network architecture are given in the Appendix.

After training, we use the encoder and bottleneck parts of the network to achieve the two-dimensional latent space representation shown in Fig. 2. The left panel of the figure shows the latent space spanned by the two bottleneck neurons with a color code corresponding to the magnetic field strength, which determines the interaction strength. We observe that all data belonging to the same magnetic field lie along a curve in the latent space. The right panel shows the same representation

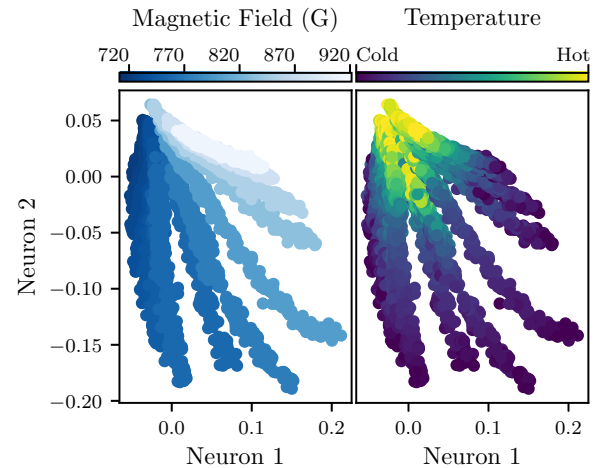


FIG. 2. Latent space representation of time-of-flight data across the crossover, in which every data point represents one compressed image. Left: The color of the data indicates the external magnetic field strength. In the latent space, the data are organized along curves of sorted interaction values. Right: Same plot as the left, but the color shows the relative temperature of data points. Since the temperatures differ for each field, the scale is normalized. The network not only learns to sort data points by interaction, but it also arranges them by temperature along each curve, with the tendency of hot clouds to converge towards a common region in the latent space.

of the latent space, however with the color map representing temperature. This depiction illustrates that the ordering along the lines of different interaction strengths is according to the temperature of the sample. It should be noted that the autoencoder does not receive any information regarding the interaction or temperature, but learns these quantities from the time-of-flight pictures alone. Additionally, the autoencoder groups thermal (hot) gases in a small concentrated area in the upper-left corner, while the (cold) superfluid gas covers a larger area at the bottom of the latent space. Hence, the neural network detects a wider range of variation in the signatures of superfluids for different interaction strengths, whereas differently interacting thermal gases are more similar to each other. For the work discussed in this paper, the reduction to two neurons at the bottleneck has proven sufficient. We found that the final loss during training, which measures the mean-squared error of the network outputs with regard to the inputs, shows no improvement for more than two neurons. Moreover, for trained networks with three bottleneck neurons, the output in the latent space is arranged on a common plane reducing the latent space to a two-dimensional distribution.

Tracing the data points from hot to cold for a given interaction strength shows that the data are not well described by a single straight line. Instead, we use piecewise linear segments to model the data and to detect a change of trend at specific temperatures. First, we average the latent space position of the  $\sim 13$  recorded images for each temperature and interaction strength setting. Next, we trim the considered temperatures for each interaction strength to a range around the determined critical temperature from our supervised analysis [18]. This approach prevents the accidental detection of trend changes at temperatures far above or below the critical temperature.

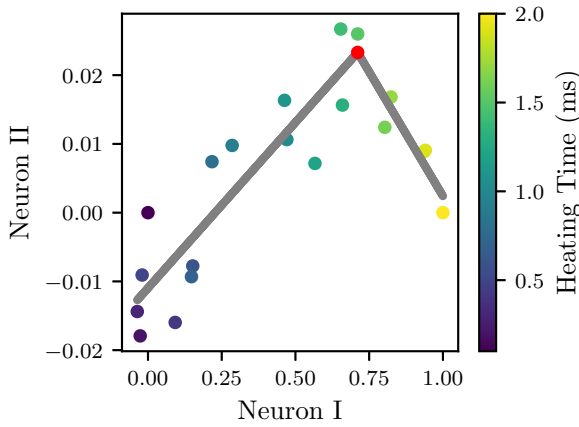


FIG. 3. Extraction of the critical temperature from the latent space representation. As an example, the extraction for data with  $1/(k_F a) \sim -0.26$  at the trend changing position is shown. Each point represents the averaged position of 13 recorded images. The gray lines display the fitted piecewise linear segments, and the red point shows the determined position of the trend change.

Around unitarity where the reduced temperature  $T/T_F$  does not change substantially at the expected critical temperature, the trimming range is chosen such that we omit temperatures at which the condensate fraction from our supervised analysis is above  $\sim 5\%$ . At fields where the reduced temperature varies strongly with the determined critical heating time and the maximally measured condensate fraction is below this limit, the chosen range comprises temperatures in a window of roughly  $\pm 25\% T/T_F$  of the previously determined critical temperatures. This range covers approximately 80–100% of the span of measured temperatures at each field. Subsequently, the latent space is shifted and linearly transformed, while preserving angles, before we apply a piecewise linear fit in order to determine the location of the trend change as shown in the transformed latent space (Neurons I and II) in Fig. 3. The fit is performed by defining a piecewise linear function with four degrees of freedom, namely two slopes and the coordinates of the trend change, which is fitted to the data via the least-squares method. We average the heating time of the three data points closest to the trend changing position, and we use our calibration of the heating measurements to infer the corresponding temperature. Since the trained network model is subject to statistical variation caused by the finite size of the shuffled data set, stochastic gradient descent during training, and randomly initialized weights of the network, we train multiple equivalent models. Next, we average the determined heating times at the trend changing positions weighted by the reciprocal variance of the previously calculated mean of the three closest data points. In total, we train 173 models and omit only one model with nonconverged loss after training. In addition, we omit every fit in which the result is at the edge of the trimmed latent space, which sums up to 33 discarded fits from a total of 2408 for all of the 172 remaining models with 14 different interactions each. These fits mainly belong to interactions around  $1/(k_F a) \sim 0.5$ , where the change of trend is least pronounced.

In Fig. 4 we compare the extracted critical temperatures to the phase boundary determined by our supervised neural

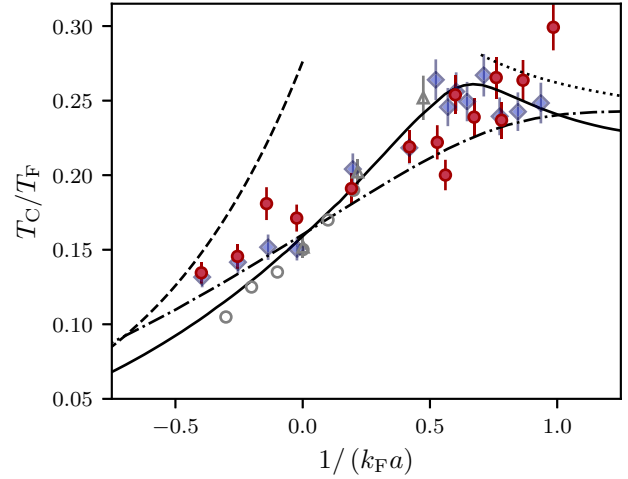


FIG. 4. Extracted critical temperatures across the BEC-BCS crossover shown as red dots. Error bars are calculated from the SE of averaging multiple trained models with differently shuffled training data sets and an estimation of the systematic error caused by nonharmonicities of the trap [18]. Blue diamonds represent the results of our supervised analysis [18]. Dashed line: BCS theory with GMB corrections; solid line: extended GMB theory [8]; dash-dotted line: theory from [2]; dotted line: interacting BEC; open triangles: quantum Monte Carlo data [4], open circles: quantum Monte Carlo data [3].

network [18] and several theories [2–4,8], finding overall very good agreement. Our analysis reveals the critical temperature in the region of  $-0.40 < 1/(k_F a) < 0.98$  and shows a striking resemblance to the extended Gor'kov-Melik-Barkhudarov (GMB) theory [8]. On the BCS side, this approach agrees well with the higher temperatures presented by Ref. [2], which exceed the suggested temperatures by quantum Monte Carlo (QMC) calculations [3]. From unitarity to the BEC side of the crossover, the temperatures match with the QMC calculations in [4]. Moreover, the determined superfluid transition temperatures agree well with the values from our previous work using supervised training of deep neural networks [18]. We therefore conclude that the change of trend in the latent space corresponds to the critical temperature, providing a new and unbiased perspective on the phase boundary in the crossover regime. Thus, this work presents another confirmation of our previous work, which in contrast does not rely on the rapid-ramp technique while retaining good accuracy.

In conclusion, we extracted the phase boundary of the superfluid phase transition in the BEC-BCS crossover directly from the momentum distribution of the fermions by application of an unsupervised autoencoder network. The network is able to identify physical quantities such as temperature and interaction strength independently of any external information, and it reveals the phase transition by a feature in the low-dimensional latent space of the data set. Our work shows that the direct detection of Fermi condensates from the momentum distribution of fermions is possible even without projection onto dimers. In this regard, this method presents an alternative approach to unveil features in experimental time-of-flight data where model-based analysis is impractical.

This work has been supported by the Deutsche Forschungsgemeinschaft (SFB/TR 185 project C6) and Cluster of Excellence Matter and Light for Quantum Computing (ML4Q) EXC 2004/1-390534769.

#### APPENDIX: AUTOENCODER NETWORK

The neural network architecture used to create the low-dimensional representation of the phase diagram is given in Table I. For training, we provide the network with 5031 centered absorption images of  $192 \times 192$  px at different temperatures and interactions across the crossover with  $\sim 13$  repetitions each. To generate the latent space, the full network is trained in a supervised way using the whole data set as both input and labels at the same time. Later, only the encoder half of the network is employed to extract the latent space representation of the data set in the bottleneck layer. This approach does not require the network to share a symmetric encoder and decoder, nor to have shared weights of mirrored layers. Every dense and convolutional layer uses the ReLU activation, except for the dense layer in the bottleneck utilizing a linear activation. The total number of parameters is 17400963. We use a batch size of 20 and train for 15 epochs with an Adam optimizer [26] and a learning rate of  $4 \times 10^{-4}$  with mean-squared error as a loss function. From the 5031 data points, we use 90% for training and 10% for

TABLE I. Neural network architecture used for the autoencoder.

Layer (type)	Output shape	Parameters
Input	(192, 192, 1)	0
2D convolutional	(192, 192, 32)	320
Max pooling	(48, 48, 32)	0
2D convolutional	(48, 48, 64)	8256
Max pooling	(16, 16, 64)	0
Flatten	(16384)	0
Dense	(512)	8389120
Dense	(512)	262656
Dense (bottleneck)	(2)	1026
Dense	(512)	1536
Dense	(512)	262656
Dense	(16384)	8404992
Reshape	(16, 16, 64)	0
2D transpose convolutional	(48, 48, 64)	36928
Batch normalization	(48, 48, 64)	256
2D transpose convolutional	(192, 192, 32)	32800
Batch normalization	(192, 192, 32)	128
2D convolutional	(192, 192, 1)	289

validation, which are randomly shuffled before being used for training. The network was realized with the TensorFlow library [25].

- [1] C. A. R. Sá de Melo, M. Randeria, and J. R. Engelbrecht, Crossover from BCS to Bose superconductivity: Transition temperature and time-dependent Ginzburg-Landau theory, *Phys. Rev. Lett.* **71**, 3202 (1993).
- [2] R. Haussmann, W. Rantner, S. Cerrito, and W. Zwerger, Thermodynamics of the BCS-BEC crossover, *Phys. Rev. A* **75**, 023610 (2007).
- [3] A. Bulgac, J. E. Drut, and P. Magierski, Quantum Monte Carlo simulations of the BCS-BEC crossover at finite temperature, *Phys. Rev. A* **78**, 023625 (2008).
- [4] E. Burovski, E. Kozik, N. Prokof'ev, B. Svistunov, and M. Troyer, Critical temperature curve in BEC-BCS crossover, *Phys. Rev. Lett.* **101**, 090402 (2008).
- [5] Y. Inada, M. Horikoshi, S. Nakajima, M. Kuwata-Gonokami, M. Ueda, and T. Mukaiyama, Critical temperature and condensate fraction of a fermion pair condensate, *Phys. Rev. Lett.* **101**, 180406 (2008).
- [6] S. Floerchinger, M. Scherer, S. Diehl, and C. Wetterich, Particle-hole fluctuations in BCS-BEC crossover, *Phys. Rev. B* **78**, 174528 (2008).
- [7] S. Floerchinger, M. M. Scherer, and C. Wetterich, Modified Fermi sphere, pairing gap, and critical temperature for the BCS-BEC crossover, *Phys. Rev. A* **81**, 063619 (2010).
- [8] L. Pisani, A. Perali, P. Pieri, and G. C. Strinati, Entanglement between pairing and screening in the Gorkov-Melik-Barkhudarov correction to the critical temperature throughout the BCS-BEC crossover, *Phys. Rev. B* **97**, 014528 (2018).
- [9] M. Pini, P. Pieri, and G. C. Strinati, Fermi gas throughout the BCS-BEC crossover: Comparative study of t-matrix approaches with various degrees of self-consistency, *Phys. Rev. B* **99**, 094502 (2019).
- [10] M. Greiner, C. A. Regal, and D. S. Jin, Emergence of a molecular Bose-Einstein condensate from a Fermi gas, *Nature (London)* **426**, 537 (2003).
- [11] M. W. Zwierlein, C. A. Stan, C. H. Schunck, S. M. F. Raupach, S. Gupta, Z. Hadzibabic, and W. Ketterle, Observation of Bose-Einstein condensation of molecules, *Phys. Rev. Lett.* **91**, 250401 (2003).
- [12] S. Jochim, M. Bartenstein, A. Altmeyer, G. Hendl, S. Riedl, C. Chin, J. Hecker Denschlag, and R. Grimm, Bose-Einstein condensation of molecules, *Science* **302**, 2101 (2003).
- [13] M. J. H. Ku, A. T. Sommer, L. W. Cheuk, and M. W. Zwierlein, Revealing the superfluid lambda transition in the universal thermodynamics of a unitary Fermi gas, *Science* **335**, 563 (2012).
- [14] C. A. Regal, M. Greiner, and D. S. Jin, Observation of resonance condensation of fermionic atom pairs, *Phys. Rev. Lett.* **92**, 040403 (2004).
- [15] M. W. Zwierlein, C. A. Stan, C. H. Schunck, S. M. F. Raupach, A. J. Kerman, and W. Ketterle, Condensation of pairs of fermionic atoms near a Feshbach resonance, *Phys. Rev. Lett.* **92**, 120403 (2004).
- [16] E. Altman and A. Vishwanath, Dynamic projection on Feshbach molecules: A probe of pairing and phase fluctuations, *Phys. Rev. Lett.* **95**, 110404 (2005).
- [17] I. Tikhonenkov, E. Pazy, Y. B. Band, M. Fleischhauer, and A. Vardi, Many-body effects on adiabatic passage through Feshbach resonances, *Phys. Rev. A* **73**, 043605 (2006).
- [18] M. Link, K. Gao, A. Kell, M. Breyer, D. Eberz, B. Rauf, and M. Köhl, Machine-learning the phase diagram of a strongly-interacting Fermi gas, *Phys. Rev. Lett.* **130**, 203401 (2023).
- [19] E. P. L. van Nieuwenburg, Y.-H. Liu, and S. D. Huber, Learning phase transitions by confusion, *Nat. Phys.* **13**, 435 (2017).

- [20] L. Wang, Discovering phase transitions with unsupervised learning, *Phys. Rev. B* **94**, 195105 (2016).
- [21] W. Hu, R. R. P. Singh, and R. T. Scalettar, Discovering phases, phase transitions, and crossovers through unsupervised machine learning: A critical examination, *Phys. Rev. E* **95**, 062122 (2017).
- [22] C. Alexandrou, A. Athenodorou, C. Chrysostomou, and S. Paul, The critical temperature of the 2D-Ising model through deep learning autoencoders, *Eur. Phys. J. B* **93**, 226 (2020).
- [23] N. Käming, A. Dawid, K. Kottmann, M. Lewenstein, K. Sengstock, A. Dauphin, and C. Weitenberg, Unsupervised machine learning of topological phase transitions from experimental data, *Mach. Learn.: Sci. Technol.* **2**, 035037 (2021).
- [24] A. Behrle, T. Harrison, J. Kombe, K. Gao, M. Link, J.-S. Bernier, C. Kollath, and M. Köhl, Higgs mode in a strongly interacting fermionic superfluid, *Nat. Phys.* **14**, 781 (2018).
- [25] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg *et al.*, TensorFlow: Large-scale machine learning on heterogeneous systems (2015), software available from [tensorflow.org](https://www.tensorflow.org).
- [26] D. P. Kingma and J. Ba, Adam: A method for stochastic optimization, [arXiv:1412.6980](https://arxiv.org/abs/1412.6980).