This means that the necessary extrapolation of the known lower rotational J transitions to the unknown higher J transitions will fail with the traditional perturbation Hamiltonian. Some progress can be made if the divergent series is resumed with the use of Padé approximants (*22*) and other more sophisticated schemes (*23*). The assignment of the hot water vapor spectrum is therefore a difficult task (*24*). The water spectrum is of fundamental importance and presents such a theoretical challenge that new techniques (*18, 25*) for the calculation of spectra are often tested with it.

REFERENCES AND NOTES

1. J. Liebert, in *Molecules in the Stellar Environment*, U. G. Jorgensen, Ed. (Springer-Verlag, Berlin, 1994), figure 1, p. 62; F. Allard, P. H. Hauschildt, S. Miller, J. Tennyson, *Astrophys. J.* **426**, L39 (1994).
2. K. H. Hinkle and T. G. Barnes, *Astrophys. J.* **227**, 923 (1979).
3. H. Spinrad and R. F. Wing, *Annu. Rev. Astron. Astrophys.* **7**, 249 (1969).
4. T. Tsuji and K. Ohnaka, in *Elementary Processes in Dense Plasmas*, S. Ichimaru and S. Ogata, Eds. (Addison-Wesley, Reading, MA, in press).
5. A. C. Cheung *et al.*, *Nature* **221**, 626 (1969).
6. J. C. Pearson, T. Anderson, E. Herbst, F. C. Delucia, P. Helminger, *Astrophys. J.* **379**, L41 (1991); T. Amano and F. Scappini, *Chem. Phys. Lett.* **182**, 93 (1991), and references therein.
7. R. J. Bray and R. E. Loughead, *Sunspots* (Dover, New York, 1979), p. 107.
8. K. Sinha, *Proc. Astron. Soc. Aust.* **9**, 32 (1991).
9. M. Geller, *A High-Resolution Atlas of the Infrared Spectrum of the Sun and the Earth Atmosphere from Space*, volume III, *Key to Identification of Solar Features* (*NASA Ref. Publ. 1224*, National Aeronautics and Space Administration, Washington, DC, 1992).
10. D. N. B. Hall, thesis, Harvard University (1970).
11. L. Wallace and W. Livingston, *An Atlas of a Dark Sunspot Umbral Spectrum from 1970 to 8640 cm^{-1} (1.16 to 5.1 μm)* (*Natl. Solar Observ. Tech. Rep. 92-001*, National Optical Astronomy Observatories, Tucson, AZ, 1992).
12. S. G. Kleinman and D. N. B. Hall, *Astrophys. J. Suppl. Ser.* **62**, 501 (1986).
13. D. N. B. Hall and R. W. Noyes, *Astrophys. J.* **175**, L95 (1972).
14. R. B. LeBlanc, J. B. White, P. F. Bernath, *J. Mol. Spectrosc.* **164**, 574 (1994).
15. J. M. Flaud, C. Camy-Peyret, J. P. Maillard, *Mol. Phys.* **32**, 499 (1976).
16. C. Camy-Peyret *et al.*, *ibid.* **33**, 1641 (1977).
17. R. A. Toth, *Appl. Opt.* **33**, 4851 (1994), and references therein.
18. O. L. Polyansky, P. Jensen, J. Tennyson, *J. Chem. Phys.* **101**, 7651 (1994), and references therein.
19. L. Wallace, W. Livingston, P. Bernath, *An Atlas of the Sunspot Spectrum from 470 to 1233 cm^{-1} (8.1 to 21 μm) and the Photospheric Spectrum from 460 to 630 cm^{-1} (16 to 22 μm)* (*Natl. Solar Observ. Tech. Rep. 1994-01*, National Optical Astronomy Observatories, Tucson, AZ, 1994).
20. D. A. Glenar, A. R. Hill, D. E. Jennings, J. W. Brault, *J. Mol. Spectrosc.* **111**, 403 (1985).
21. J. M. Campbell, D. Klapstein, M. Dulick, P. F. Bernath, L. Wallace, in preparation.
22. O. L. Polyansky, *J. Mol. Spectrosc.* **112**, 79 (1985).
23. V. G. Tyuterev, *ibid.* **151**, 97 (1992).
24. O. L. Polyansky, J. Busler, B. Guo, K. Zhang, P. Bernath, in preparation.
25. U. G. Jorgensen and P. Jensen, *J. Mol. Spectrosc.* **161**, 219 (1993).
26. W. Livingston, *Nature* **350**, 45 (1991).

27. The National Optical Astronomy Observatories are operated by the Association of Universities for Research in Astronomy under cooperative agreement with the National Science Foundation. We thank T. Tsuji for a copy of (*4*) in advance of publication. This work was supported by the Natural Sciences and Engineering Research Council of Canada. This work was supported in part by the Petroleum Research Fund. Some support was also provided by the National Aeronautics and Space Administration Laboratory Astrophysics Program.

3 February 1995; accepted 17 March 1995

The "Wake-Sleep" Algorithm for Unsupervised Neural Networks

Geoffrey E. Hinton,* Peter Dayan, Brendan J. Frey, Radford M. Neal

An unsupervised learning algorithm for a multilayer network of stochastic neurons is described. Bottom-up "recognition" connections convert the input into representations in successive hidden layers, and top-down "generative" connections reconstruct the representation in one layer from the representation in the layer above. In the "wake" phase, neurons are driven by recognition connections, and generative connections are adapted to increase the probability that they would reconstruct the correct activity vector in the layer below. In the "sleep" phase, neurons are driven by generative connections, and recognition connections are adapted to increase the probability that they would produce the correct activity vector in the layer above.

Supervised learning algorithms for multilayer neural networks face two problems: They require a teacher to specify the desired output of the network, and they require some method of communicating error information to all of the connections. The wake-sleep algorithm avoids both of these problems. When there is no external teaching signal to be matched, some other goal is required to force the hidden units to extract underlying structure. In the wake-sleep algorithm, the goal is to learn representations that are economical to describe but allow the input to be reconstructed accurately. We can quantify this goal by imagining a communication game in which each vector of raw sensory inputs is communicated to a receiver by first sending its hidden representation and then sending the difference between the input vector and its top-down reconstruction from the hidden representation. The aim of learning is to minimize the "description length," which is the total number of bits that would be required to communicate the input vectors in this way (*1*). No communication actually takes place, but minimizing the description length that would be required forces the network to learn economical representations that capture the underlying regularities in the data (*2*).

The neural network has two quite different sets of connections. The bottom-up "recognition" connections are used to convert the input vector into a representation in one or more layers of hidden units. The top-down "generative" connections are then used to reconstruct an approximation of the input vector from its underlying representation. The training algorithm for these two sets of connections can be used with many different types of stochastic neurons, but for simplicity, we use only stochastic binary units that have states of 1 or 0. The state of unit v is s_v, and the probability that it is on is

$$\mathrm{Prob}(s_v = 1) = \frac{1}{1 + \exp(-b_v - \sum_u s_u w_{uv})} \qquad (1)$$

where b_v is the bias of the unit and w_{uv} is the weight on a connection from unit u. Sometimes the units are driven by the generative weights, and at other times they are driven by the recognition weights, but the same equation is used in both cases (Fig. 1).

In the "wake" phase, the units are driven bottom-up with the recognition weights; this produces a representation of the input vector in the first hidden layer, a representation of this representation in the second hidden layer, and so on. All of these layers of representation combined are called the "total representation" of the input, and the binary state of each hidden unit j in the total representation α is s_j^{α}. This total representation could be used to communicate the input vector d to a receiver. According to Shannon's coding theorem, it requires $-\log r$ bits to communicate an event that has probability r under a distribution agreed upon by the sender and receiver. We assume that the receiver knows the top-down generative weights (*3*), so that these can be used to create the agreed probability distributions required for communication. First, the activ-

Department of Computer Science, University of Toronto, 6 King's College Road, Toronto, M5S 1A4, Ontario, Canada.

*To whom correspondence should be addressed.

ity of each unit k in the top hidden layer is communicated using the distribution (p_k^α, $1 - p_k^\alpha$), which is obtained by applying Eq. 1 to the single generative bias weight of unit k. Then the activities of the units in each lower layer are communicated using the distribution (p_j^α, $1 - p_j^\alpha$) obtained by applying Eq. 1 to the already communicated activities in the layer above, s_k^α, and to the generative weights, w_{kj}. The description length of the binary state of unit j is

$$C(s_j^\alpha) = -s_j^\alpha \log p_j^\alpha - (1 - s_j^\alpha)\log(1 - p_j^\alpha) \qquad (2)$$

The description length for input vector d using the total representation α is simply the cost of describing all the hidden states in all the hidden layers plus the cost of describing the input vector given the hidden states

$$\begin{aligned} C(\alpha,d) &= C(\alpha) + C(d\,|\,\alpha) \\ &= \sum_{\ell \in L} \sum_{j \in \ell} C(s_j^\alpha) + \sum_i C(s_i^d\,|\,\alpha) \qquad (3) \end{aligned}$$

where ℓ is an index over the L layers of hidden units and i is an index over the input units that have states s_i^d.

Because the hidden units are stochastic, an input vector will not always be represented in the same way. In the wake phase, the recognition weights determine a conditional probability distribution $Q(\cdot\,|\,d)$ over total representations. Nevertheless, if the recognition weights are fixed, there is a very simple, on-line method of modifying the generative weights to minimize the expected cost $\Sigma_\alpha Q(\alpha\,|\,d)C(\alpha,d)$ of describing the input vector with a stochastically chosen total representation. After the recognition weights are used to choose a total representation, each generative weight is adjusted in proportion to the derivative of Eq. 3 by use of the purely local delta rule

$$\Delta w_{kj} = \epsilon s_k^\alpha (s_j^\alpha - p_j^\alpha) \qquad (4)$$

where ϵ is a learning rate. Although the units are driven by the recognition weights, it is only the generative weights that learn in the wake phase. The learning makes each layer of the total representation better at reconstructing the activities in the layer below.

It seems obvious that the recognition weights should be adjusted to maximize the probability of picking the α that minimizes $C(\alpha,d)$. But this is incorrect. When there are many alternative ways of describing an input vector, it is possible to design a stochastic coding scheme that takes advantage of the entropy across alternative descriptions (*1*). The cost is then

$$C(d) = \sum_\alpha Q(\alpha|d)C(\alpha,d) - \left[-\sum_\alpha Q(\alpha\,|\,d)\log Q(\alpha\,|\,d)\right] \qquad (5)$$

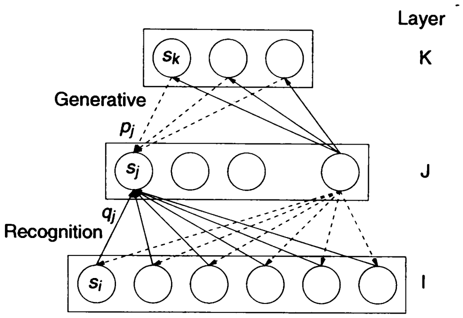

Fig. 1. A three-layer Helmholtz machine. The bottom layer represents the raw sensory inputs. Units in layers I, J, and K are completely interconnected with recognition (solid lines) and generative (dotted lines) connections. The binary activity of unit j in layer J is s_j. The quantity q_j is determined by the recognition weights, and p_j is determined by the generative weights. When the units are driven bottom-up, the probability that $s_j = 1$ is q_j; when they are driven top-down, the probability is p_j.

The second term on the right is the entropy of the distribution that the recognition weights assign to the various alternative representations. If, for example, there are two alternative representations, each of which costs 4 bits, the combined cost is only 3 bits provided we use the two alternatives with equal probability (*4*). It is precisely analogous to the way in which the energies of the alternative states of a physical system are combined to yield the Helmholtz free energy of the system. As in physics, $C(d)$ is minimized when the probabilities of the alternatives are exponentially related to their costs by the Boltzmann distribution (at a temperature of 1)

$$P(\alpha\,|\,d) = \frac{\exp[-C(\alpha,d)]}{\sum_\beta \exp[-C(\beta,d)]} \qquad (6)$$

So, rather than adjusting the recognition weights to focus all of the probability on the lowest cost representation, we should try to make the recognition distribution $Q(\cdot\,|\,d)$ as similar as possible to the Boltzmann distribution $P(\cdot\,|\,d)$, which is the posterior distribution over representations given the data and given the network's generative model. It is exponentially expensive to compute $P(\cdot\,|\,d)$ exactly (*5*), but there is a simple way of getting approximately correct target states for the hidden units in order to train the distribution $Q(\cdot\,|\,d)$ produced by the bottom-up recognition weights.

We turn off the recognition weights and drive all of the units in the network with the generative weights, starting at the topmost hidden layer and working down all the way to the input units. Because the units are stochastic, repeating this process typically produces many different "fantasy" vectors on the input units. These fantasies provide an unbiased sample of the network's generative model of the world. Having produced a fantasy, we then adjust the recognition weights to maximize the logarithm of the probability of recovering the hidden activities that actually caused the fantasy

$$\Delta w_{jk} = \epsilon s_j^\gamma (s_k^\gamma - q_k^\gamma) \qquad (7)$$

where γ specifies the states of both the hidden units and the input units for a particular fantasy and q_k^γ is the probability that unit k would be turned on by the recognition weights operating on the binary activities s_j^γ in the layer below (*6*). We call this the "sleep" phase of the algorithm. Like the wake phase, it uses only locally available information. A potential drawback of the sleep phase is that we would like the recognition weights to be good at recovering the true causes for the training data but the sleep phase optimizes the recognition weights for fantasy data. Early in the learning, fantasies will have a quite different distribution than the training data.

The distribution $Q(\cdot\,|\,d)$ produced by the recognition weights is a factorial distribution in each hidden layer because the recognition weights produce stochastic states of units within a hidden layer that are conditionally independent, given the states in the layer below. It is natural to use factorial distributions in a neural net because it allows the probability distribution over the 2^n alternative hidden representations to be specified with n numbers instead of $2^n - 1$. This simplification, however, will typically make it impossible for the distribution $Q(\cdot\,|\,d)$ to exactly match the posterior distribution $P(\cdot\,|\,d)$ in Eq. 6. It makes it impossible, for example, to capture "explaining away" effects where the activity vector in one layer can be economically explained by activating either unit a or unit b in the layer above but not by activating both of them.

The restriction of $Q(\cdot\,|\,d)$ to a factorial distribution is a potentially very serious limitation. The reason it is not a fatal flaw is that the wake phase of the algorithm adapts the generative weights so as to make $P(\cdot\,|\,d)$ close to $Q(\cdot\,|\,d)$, thus limiting the loss caused by the inability of $Q(\cdot\,|\,d)$ to model nonfactorial distributions. To see why this effect occurs, it is helpful to rewrite Eq. 5 in a different form

$$\begin{aligned} C(d) = &\sum_\alpha P(\alpha\,|\,d)C(\alpha,d) \\ &- \left(-\sum_\alpha P(\alpha\,|\,d)\log P(\alpha\,|\,d)\right) \\ &+ \sum_\alpha Q(\alpha\,|\,d)\log\frac{Q(\alpha\,|\,d)}{P(\alpha\,|\,d)} \qquad (8) \end{aligned}$$

The first two terms on the right in Eq. 8 are exactly $-\log P(d)$ under the current generative model. The last term, which cannot be

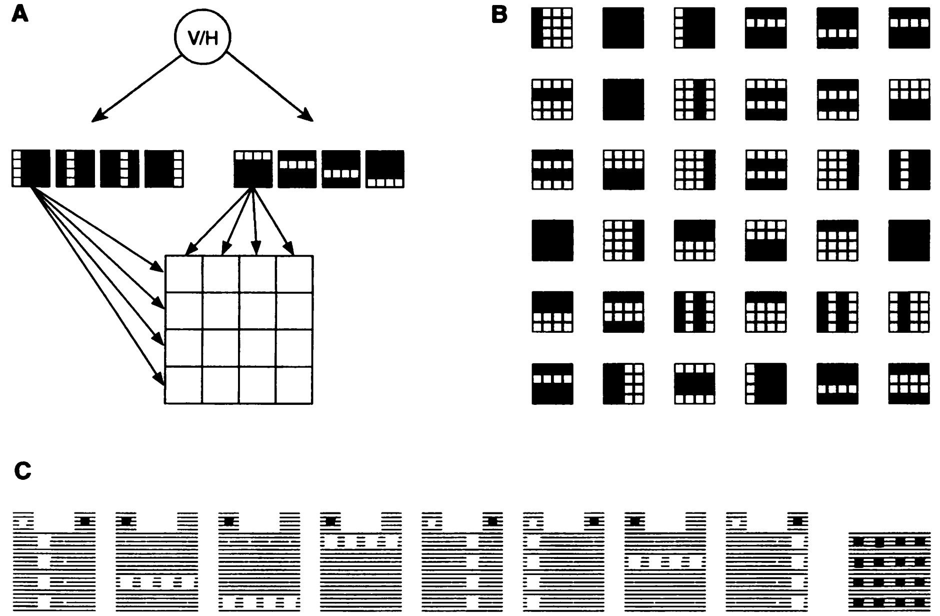

Fig. 2. (**A**) A generative model for 4 × 4 images. The top level decides whether to use vertical (V) or horizontal (H) bars. The next level decides whether each possible bar of the chosen orientation should be present in the image. (**B**) A sample of the images produced by the model in (A) with the ambiguous all-white images removed. A neural net with 16 input units, 8 units in the first hidden layer, and 1 hidden unit in the second hidden layer was trained on 2 × 10^6 random examples produced by the generative model. After training (*17*), the probability distribution produced in the sleep phase was almost exactly correct. (**C**) The generative weights to and from the 8 units in the first hidden layer. Positive weights are white, negative weights are black, and the area is proportional to the magnitude. The largest weight shown is 14.1. The generative bias of the unit is shown on the top right of each block, and its generative weight from the single unit in the layer above is shown on the top left. The right-most block shows the generative biases of the input units. To encourage an easily interpretable solution, the generative weights to the input units were constrained to be positive. If they are allowed to go negative, the algorithm finds solutions that produce the correct distribution but in a much more complicated way, and it requires more units in the second hidden layer.

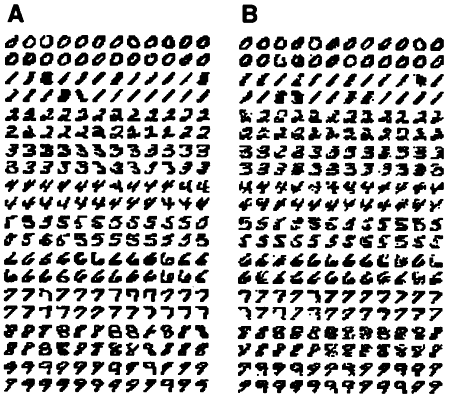

Fig. 3. Handwritten digits were normalized and quantized to produce 8 × 8 binary images. In (**A**) are shown 24 examples of each digit. A separate network was trained on each digit class, and 24 fantasies from each network are shown in (**B**). The variations within each digit class are modeled quite well. The error rate was 4.8% when new test images were classified by choosing the network that minimized the description length of the image. On the same data, nearest neighbor classification gave 6.7% errors, and back-propagation training of a single supervised net with 10 output units and one hidden layer gave a minimum of 5.6% errors even when we used the test data to optimize the number of hidden units, the training time, and the amount of weight decay (*7*).

negative, is the Kullback-Leibler divergence between $Q(\cdot|d)$ and $P(\cdot|d)$, which is the amount by which the description length with $Q(\cdot|d)$ exceeds $-\log P(d)$. Thus, for two generative models that assign equal probability to d, minimizing Eq. 8 with respect to the generative weights will tend to favor the model whose posterior distribution is most similar to $Q(\cdot|d)$. Within the available space of generative models, the wake phase seeks out those models that give rise to posterior distributions that are approximately factorial.

Because we are making several approximations, the algorithm must be evaluated by its performance. Figure 2 shows that it can learn the correct multilayer generative model for a simple toy problem. Moreover, after learning, the Kullback-Leibler divergence in Eq. 8 is only 0.08 bit, which indicates that this term has forced a solution in which the generative model has an almost perfectly factorial posterior.

We also tested the algorithm on two quantitative aspects of its capacity to build models of images of highly variable handwritten digits (Fig. 3A). Learning 10 different models, one for each digit, we were able to recognize new digits accurately by seeing which models gave the most economical descriptions of them. Figure 3B shows that after the algorithm has learned a digit model, the fantasies generated by the network are very similar to the real data. We also trained a single large network on all the digits and confirmed that it compressed new digits almost as well as did these 10 digit-specific networks, and nearly twice as well as a naïve code (*7*).

Two of the most widely used unsupervised training algorithms for neural networks are principal components analysis and competitive learning (sometimes called vector quantization or clustering). Both can be viewed as special cases of the minimum description length approach, in which there is only one hidden layer and it is unnecessary to distinguish between the recognition and generative weights because they are always the same (*8*). Other learning schemes have been proposed that use separate feed-forward and feedback weights (*9–12*). By contrast with adaptive resonance theory (*9*), the counter-streams model (*10*), and the algorithm of Kawato *et al.* (*11*), the wake-sleep algorithm treats the problem of unsupervised learning as statistical—one of fitting a generative model that accurately captures the structure in the input examples. Kawato's model is couched in terms of forward and inverse models (*13*), which constitutes an alternative way to look at our generative and recognition models. The wake-sleep algorithm is closest in spirit to Barlow's ideas about invertible factorial representations (*14*) and Mumford's proposals (*12*) for mapping Grenander's generative model approach (*15*) onto the brain.

The minimum description length approach to unsupervised learning was developed to improve the pattern recognition abilities of artificial neural networks, but the simplicity of the wake-sleep learning algorithm makes it biologically interesting. For example, Hasselmo and Bower (*16*) have suggested that cholinergic inputs to the cortex may modulate the degree of feed-forward control of ongoing activity. By a curious coincidence, the idea that the perceptual system uses generative models was advocated by Helmholtz, so we call any neural network that fits a generative model to data by minimizing the free energy in Eq. 5 a "Helmholtz machine."

REFERENCES AND NOTES

1. J. Rissanen, *Stochastic Complexity in Statistical Inquiry* (World Scientific, Singapore, 1989).
2. The description length can be viewed as an upper bound on the negative logarithm of the probability of the data given the network's generative model, so

this approach is closely related to maximum likelihood methods of fitting models to data.

3. The number of bits required to specify the generative weights should also be included in the description length (*1*), but we currently ignore it.
4. G. E. Hinton and R. S. Zemel, in *Advances in Neural Information Processing Systems* J. D. Cowan, G. Tesauro, J. Alspector, Eds. (Kaufmann, San Mateo, CA, 1994), vol. 6, pp. 3–10.
5. An unbiased estimate of the exact gradient is easy to obtain, but the noise in this estimate increases with the size of the network. Alternatively, a mean-field approximation can be made to the stochastic recognition model and the error derivatives can then be computed by a back-propagation process (P. Dayan, G. E. Hinton, R. M. Neal, R. S. Zemel, *Neural Comput.*, in press).
6. This performs stochastic steepest descent in the Kullback-Leibler divergences
$$\sum_{\ell \in L} \sum_{j \in \ell} p_j^{\gamma} \log(p_j^{\gamma}/q_j^{\gamma}) + (1 - p_j^{\gamma})\log[(1 - p_j^{\gamma})/(1 - q_j^{\gamma})] \qquad (9)$$
The cost function in Eq. 5 contains the same terms but with p and q interchanged, leading to an approximation error that is equal to the asymmetry of the Kullback-Leibler divergences.
7. The training data was 700 examples of each digit from the CEDAR CDROM 1 made available by the U.S. Postal Service Office of Advanced Technology. Starting with the input layer, each of the 10 digit-specific networks had a 64-16-16-4 architecture. All weights were started at 0, and the learning rate on all connections was 0.01. Training involved 500 sweeps through the 700 examples. For testing, each net was run 10 times to estimate the expected description length of the image. The single network trained on all the digits had a 64-32-32-16 architecture and was also trained for 500 sweeps through the 7000 examples. If just the base rates of the individual pixels in the input were used, the average code length per test digit was 59.0 bits; if the 10 digit-specific networks were used, the average length was 33.8 bits, including the 3.3 bits required to specify which model was being used; if the single network was used, the average length was 37.3 bits.
8. In principal components analysis, the hidden representation vector is a linear function of the input vector and the aim is to minimize the squared reconstruction error. From a description length perspective, the cost of describing the hidden activities is ignored, so that only the cost of describing the reconstruction errors needs to be minimized. If these errors are coded with the use of a zero-mean Gaussian, the cost of describing them is proportional to their squared values. In competitive learning, only the hidden unit whose weight vector is most similar to the input vector is activated. The reconstruction is just the weight vector of the winning hidden unit, and so minimizing the squared distance between the input vector and the weight vector of the winning hidden unit minimizes the description length of the reconstruction error.
9. G. Carpenter and S. Grossberg, *Comput. Vision Graphics Image Process.* **37**, 54 (1987).
10. S. Ullman, in *Large-Scale Theories of the Cortex*, C. Koch and J. Davis, Eds. (MIT Press, Cambridge, MA, 1994), pp. 257–270.
11. M. Kawato, H. Hayakama, T. Inui, *Network: Comput. Neural Syst.* **4**, 415 (1993).
12. D. Mumford, in *Large-Scale Theories of the Cortex*, C. Koch and J. Davis, Eds. (MIT Press, Cambridge, MA, 1994), pp. 125–152.
13. M. I. Jordan and D. E. Rumelhart, *Cognitive Sci.* **16**, 307 (1992).
14. H. B. Barlow, *Neural Comput.* **1**, 295 (1989).
15. U. Grenander, *Lectures in Pattern Theory I, II, and III: Pattern Analysis, Pattern Synthesis, and Regular Structures*, (Springer-Verlag, Berlin, 1976–1981).
16. M. E. Hasselmo and J. M. Bower, *Trends Neurosci.* **16**, 218 (1993).
17. The learning rates were 0.2 for the generative and recognition weights to and from the input units and 0.001 for the other weights. The generative biases of the first hidden layer started at −3.00, and all other weights started at 0.0. The final asymmetric divergence between the network's generative model and the real data was 0.10 bit. The penalty term in Eq. 8 was 0.08 bit.
18. This research was supported by Canadian federal grants from the National Sciences and Engineering Research Council and the Institute for Robotic and Intelligent Systems and an Ontario grant from the Information Technology Research Centre. G.E.H. is the Noranda Fellow of the Canadian Institute for Advanced Research.

29 November 1994; accepted 4 April 1995

Crack-Like Sources of Dislocation Nucleation and Multiplication in Thin Films

D. E. Jesson,* K. M. Chen, S. J. Pennycook, T. Thundat, R. J. Warmack

With the combination of the height sensitivity of atomic force microscopy and the strain sensitivity of transmission electron microscopy, it is shown that near singular stress concentrations can develop naturally in strained epitaxial films. These crack-like instabilities are identified as the sources of dislocation nucleation and multiplication in films of high misfit. This link between morphological instability and dislocation nucleation provides a method for studying the basic micromechanisms that determine the strength and mechanical properties of materials.

Dislocation nucleation in thin films is of considerable scientific and technological importance in research areas ranging from the transport properties of superconducting layers to the regulation of electrical and optical properties in semiconductor devices. The mechanism by which the first dislocations nucleate in a continuous thin film has been a central and unresolved issue of strained-layer epitaxy. It is known that misfit stress in thin films can be relieved by the introduction of either a nonplanar surface morphology (*1–7*) or misfit dislocations (*8–10*), but the connection and relative importance of these mechanisms has not been explored. Furthermore, the identification of dislocation sources and multiplication mechanisms presents an outstanding experimental challenge.

Here we study strain relaxation in the technologically important Si-Ge system, which illustrates the general physical principles governing the growth of strained thin films. Our approach is to combine atomic force microscopy (AFM) with transmission electron microscopy (TEM) to provide complementary local height and strain information. This procedure reveals that crack-like surface instabilities develop spontaneously and act as the sources of misfit dislocations in strained thin films. These observations connect the previously disparate fields of morphological instability and dislocation nucleation through the natural framework of fracture mechanics.

To examine the connection between dislocation nucleation and morphological instability at high misfits, it is necessary to study the critical transition regime between a coherent (highly stressed) and dislocated (partially relaxed) film. This transition, which occupies only a very small region of the enormous phase space of deposition variables, was achieved in two stages. Initially, a 10-nm-thick $Si_{0.5}Ge_{0.5}$ alloy layer was deposited on Si(001) by molecular beam epitaxy at 400°C to create a dislocation-free film associated with a nominally planar surface. The morphological instability of this surface was demonstrated by a 1-min in situ anneal at 560°C, during which a surface ripple morphology develops, as shown in Fig. 1, A and B. The ripple consists of island-like features that align along the elastically soft [100] and [010] directions, resulting in an arrangement of orthogonal domains. Typically, the islands are 15 nm high and 100 nm in diameter, with a strong tendency to facet along {501} planes. The formation of these low-energy planes would seem to stabilize the misfit-induced morphological instability, resulting in a network of significantly stressed valleys at island intersections located ~4 nm above the alloy-substrate interface. We would emphasize that this situation results from the instability of a planar film surface and is appreciably different from the case in which the film grows initially by means of isolated islands (*11*) or fractures to create islands (*12*).

The AFM image in Fig. 1A directly links this critical point in morphological instability with the onset of dislocation nucleation (*13*). Although island heights range from 13 to 18 nm, the dislocations (arrowed) are always associated with the tallest islands.

D. E. Jesson, K. M. Chen, S. J. Pennycook, Solid State Division, Oak Ridge National Laboratory, Oak Ridge, TN 37831, USA.
T. Thundat and R. J. Warmack, Health Sciences Research Division, Oak Ridge National Laboratory, Oak Ridge, TN 37831, USA.

*To whom correspondence should be addressed.

Science

The "wake-sleep" algorithm for unsupervised neural networks

GE Hinton, P Dayan, BJ Frey and RM Neal

Science **268** (5214), 1158-1161.
DOI: 10.1126/science.7761831