

Information Density and Cortical Magnification Factors

M D Plumbley

Division of Engineering, King's College London
Strand, London WC2R 2LS, UK

Mark.Plumbley@kcl.ac.uk

Abstract

Many neural models have been suggested for the development of feature maps in cortical areas. Undoubtedly the most popular model is the Kohonen self-organizing map (SOM). Once the map has been learned, this network uses a competitive winner-take-all (WTA) approach to choose a single 'best' output neuron on a (typically) 2D grid for each presented input pattern. Cortical maps in biological organisms are known to have an expanded representation in some areas. For example, the fovea has an expanded representation in the visual cortex, and the auditory cortex of the bat has an expanded representation around its range-finding frequency. It is well known that an apparently similar effect can be observed in the Kohonen SOM. If certain input patterns occur with high probability, they tend to have a correspondingly larger number of neurons representing them: the expansion of their representation on the map is often referred to as the 'Magnification Factor'. In this paper, we suggest an alternative approach where information is represented in a distributed manner over the network, but which has the same magnification factor effect. In this approach, the map attempts to make the *information density* uniform, rather than the probability density.

1 Introduction

It is well known that sensory information on a number of modalities is arranged in a spatial 'map' in biological brains, such that information from similar sensors arrives close together in the cortex. For example, much visual sensory information is represented in a map in the primary visual cortex which is arranged roughly as the image coming in through the eyes. The human somesthetic cortex receives touch information arranged so that sensors which are close to each other on the body tend to be represented in close areas in the cortex. Similarly the bat auditory cortex receives information arranged by frequency, so that close frequencies produce a similar response.

A common feature of these feature maps is that the representation scale is non-uniform: some areas are magnified when compared with others. For example, the area of the visual cortex which corresponds to the fovea is magnified much more than that for peripheral vision (in terms of angle on the retina). The cortical area given over to touch sensors in the lips and fingers is proportionally much greater than that given over to the back. The bat has a much magnified representation for frequencies around its echo-location frequency than elsewhere [1].

These magnified areas of feature maps all seem to correspond to receptors which are proportionally more important than others. Information from the fovea

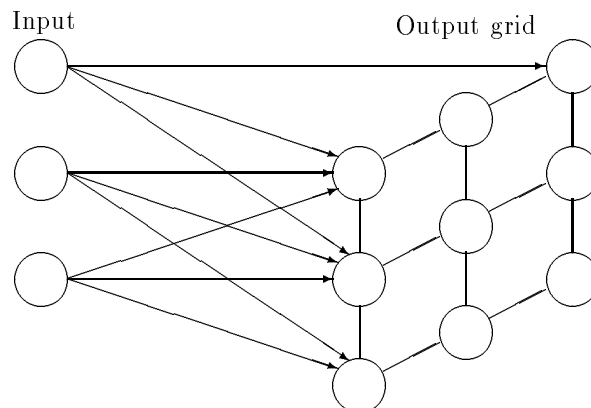


Figure 1: Kohonen SOFM structure

is required for discernment of detail in an object; information from the fingertips or lips is needed for fine manipulation of tools or food; while the detailed frequency differences around the bat’s echo-location frequency are important if it is to find and catch its prey.

The appearance of feature maps, and this magnification property, is so widespread in biological systems, that it seems reasonable to suppose that it performs a useful function. If we can understand why feature maps might have such a magnification property, it may help us to construct effective artificial sensory systems in the future.

2 Artificial neural feature maps

Inspired by these biological feature maps, Kohonen introduced his now well-known Self-Organized Feature Map (SOFM) [5], also known as the Self-Organizing Map (SOM) or ‘Kohonen Map’. While a number of modifications to the structure and learning algorithm of this model have been suggested by a number of authors, the same underlying principles continue to be used in applications today.

The SOFM is structured as follows. It is a 2-layer neural network, with one layer of input neurons, and one layer of output neurons. The input neurons are set to component values of the current input pattern vector \mathbf{x} . The output layer of neurons is a winner-take-all (WTA) network, so that one and only one output neuron is ‘on’ for each input pattern: the remainder are ‘off’. Each output neuron has a *target* input, represented by a weight vector, to which it will respond most strongly: the neuron whose target is closest to the current input pattern gets to turn on.

The output neurons are arranged on a spatial grid (often 2-dimensional and rectangular), which is used during the network’s learning process (Fig. 1). Whenever an input pattern is presented, the weight vector for the ‘winning’ output neuron is moved a little towards that pattern, so that its weight vector becomes a little more similar to the current input. Also, the weight vectors for neurons which are ‘close’ to the winner on the spatial grid are also moved towards the new input (‘closeness’ is defined by a neighbourhood measure that is initially large but decreases during training). In this way, spatially neighbouring neurons on the grid are forced to respond to similar input vectors. For more details of the structure and learning process, see e.g. [4].

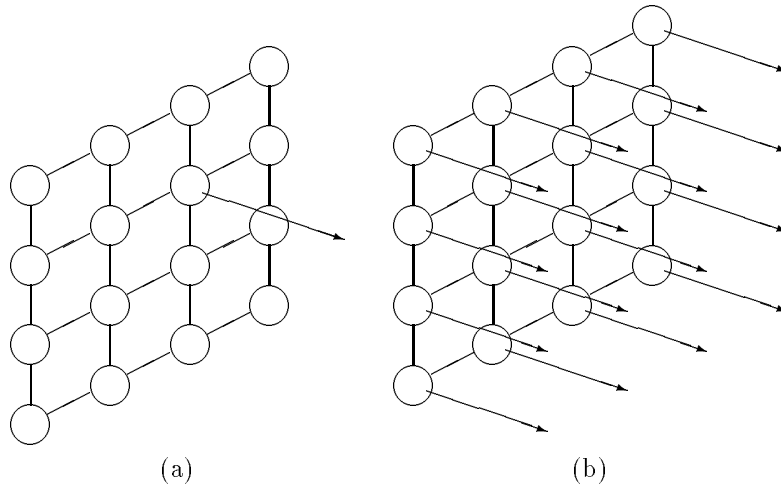


Figure 2: Information is transmitted through a winner-take-all map, such as the Kohonen Map (a), by the identity of a single ‘on’ neuron. We are searching for an alternative model where information is transmitted in parallel (b).

Once learned, the SOFM has a number of features that are similar to biological feature maps. It is composed of a number of simple neuron-like processing elements, and similar input patterns tend to be represented as a response in spatially close areas of the map. It also has a magnification property: areas of the map which correspond to input patterns which occur with high probability density $p(\mathbf{x})$ are given a proportionally larger representation.

A recent alternative feature map learning algorithm, the Generative Topographic Map (GTM) [3] is designed specifically to model the input data directly, and give a direct probability density model.

The result is that these feature maps extract a low-dimensional representation of the input patterns, where each input vector corresponds to a particular position on the map. This has a number of advantages in real applications, since it allows high-dimensional data to be projected onto a 2-D grid and thus easily visualized. The approach is also justifiable from coding theory, under the assumption that the identity of the exact position on the grid may become corrupted to a neighbouring position during transmission of the position information to later processing stages [8].

However, the winner-take-all (WTA) mechanism means that only one node is on at any time, no matter how many nodes are in the grid. Thus for a 2-D grid of 10×10 neurons the same information (a choice of one from 100 different 2-D vectors) is being spread around 100 neurons that could be represented by just two neurons with 10 just-noticeable-difference (JND) levels.

Of course, it is still possible that biology acts in this way. Perhaps it is particularly robust to noise. However, here we shall consider a possible alternative approach, allowing feature maps to deal with a graded, parallel stream of information across the map (Fig. 2).

Here we shall consider *information* density, rather than *probability* density, as a possible driving force behind the emergence of magnification factors in neural feature maps.

3 Information theory and information density

Over recent years, a number of authors have been interested in the use of information theory to propose or analyze neural network learning algorithms. Linsker [6] proposed that an unsupervised neural network should attempt to maximise the information transmitted across it: his *Infomax* principal. Linsker also developed a learning rule for winner-take-all maps from his Infomax principal [7]. Atick and Redlich [2, ?] found that early visual processing in the retina is very close to that predicted from attempting to optimize the transmitted information in the visual sensory input, through minimizing the redundancy. The current author used information theory to analyze and develop learning algorithms for decorrelating networks [10], and analyze the convergence of principal subspace networks [11].

The central concepts from information theory are *entropy* $H(X)$ of a random variable (RV) X and the *mutual information* $I(X, Y)$ between two RVs X and Y [12].

Entropy is a measure of uncertainty in a RV. For a discrete RV which taking values $X = x_i$ with probabilities p_i it is defined as

$$H(X) \triangleq - \sum_i p_i \log p_i \quad (1)$$

using $p \log p = 0$ when $p = 0$. This entropy $H(X)$ is always non-negative, is maximized when the probabilities p_i are all equal (maximum uncertainty), and zero (i.e. minimized) when $p_i = 1$ for some i and zero for all others. In the latter case, the value of the RV is known with probability 1, and there is no uncertainty.

This definition can be extended to two or more RVs. For example, if the pair of RVs X and Y take on the values $X = x_i$ and $Y = y_j$ with joint probability p_{ij} , then the *joint entropy* will be

$$H(X, Y) \triangleq - \sum_{ij} p_{ij} \log p_{ij}, \quad (2)$$

with the obvious generalization to n RVs. The joint entropy is never greater than the sum of the individual entropies: i.e.

$$H(X, Y) \leq H(X) + H(Y) \quad (3)$$

where equality holds iff X and Y are independent.

From joint entropy we can also define *conditional entropy* of X given Y as follows:

$$H(X|Y) \triangleq - \sum_{ij} p_{ij} \log \frac{p_{ij}}{p_j} \quad (4)$$

which is the uncertainty remaining in X if the values of Y are known. For any X and Y , we have $H(X|Y) < H(X)$.

Mutual information, $I(X, Y)$, can then be defined from entropy as

$$I(X, Y) \triangleq H(X) - H(X|Y) \quad (5)$$

and can thus be viewed as the reduction in uncertainty in the RV X obtained from the act of observing the RV Y . It can easily be shown that $I(X, Y)$ is symmetrical in X and Y , i.e. $I(X, Y) = I(Y, X)$. For more detail on how information theory has been applied to neural networks, see e.g. [4].

4 Properties of information density and information distribution

We have suggested that we could consider *information density* as a possible factor driving the formation of feature maps. Before we can proceed, we need to define what this information density should be, and how we might measure it.

What we would like from an information density measure is to be able to use it to answer to the following question: ‘How is the information distributed across a feature map?’. We hope to be able to define an information ‘density’ measure over either continuous or discrete domains. For the case of finite domains, such as feature maps with a finite number of output neurons, we will call this the *information distribution* by analogy with probability distribution. Like probability distribution or density, we are looking for a measure which will allow us to find the contribution towards the total by adding (or integrating) the individual distributed amounts.

Let us start by defining the entropy distribution and build information distribution from there.

The first property we propose that an entropy distribution measure should have is *summability*. We should be able to integrate entropy distribution across a domain to obtain the total entropy represented across that domain. In the case of a map with a finite number of output neurons, we should simply be able to sum the entropy distribution assigned to each of the neurons in the map to get the total amount of information distributed across the map.

Considering a ‘map’ with just two outputs X and Y , since we know that $H(X, Y) < H(X) + H(Y)$ if X and Y are not independent, then the portion of total entropy that is distributed to X and Y will be less than the entropy of each point calculated alone.

This leads to the second property we would like: that of *symmetry*. If some of the total entropy in the output is represented equally at two points (e.g. if their responses are completely correlated then the portion of the entropy distribution assigned to these two points should also be equal. For example, if the response of two neurons was always exactly the same, we should expect them to have the same information density. In other words, swapping indices should not change the measure.

A final property we would like is *accountability*. If one of the variables in the map contains all the entropy (or information), we would like the entropy distribution measure to reflect that. So if $H(X, Y) = H(X)$ while $H(Y) = 0$, then our entropy distribution should show that X is delivering all of the total entropy.

There are entropy measures, mean entropy and conditional entropy, used with random vectors and stochastic processes which have similar properties to the ones we require [9].

Let us start with mean entropy. Suppose we have a random vector $\mathbf{X} = (X_1, \dots, X_m)$. The *mean entropy* for each component X_i of \mathbf{X} is simply

$$\bar{H} = H(X_1, \dots, X_m)/m. \quad (6)$$

How does this compare with the properties we are looking for?

Mean entropy has the summability property: we simply sum the mean entropy for the m components X_1 to X_m to get the total $m\bar{H} = H(X_1, \dots, X_m)$ as required. It certainly has the symmetry property, but does not have the accountability property, since the amount allocated to outputs is always equal.

Now let us consider the conditional entropy of a stochastic process. Suppose we have a discrete-time stochastic process X_n such that at time step n we receive

the RV X_n , and X_{n-1}, \dots, X_1 have already been received. Then the *conditional entropy* of X_n is

$$H_c(X_n) = H(X_n|X_{n-1}, \dots, X_1) \quad (7)$$

i.e. the entropy of the current RV given all the previous RVs.

Conditional entropy has the summability property: it is straightforward to show that

$$H(X_1) + H(X_2|X_1) + \dots + H(X_n|X_{n-1}, \dots, X_1) = H(X_n, X_{n-1}, \dots, X_1) \quad (8)$$

which is the total entropy.

It also has an accountability property, since $H_c(X_n) < H(X_n)$ so a RV with zero entropy will lead to a conditional entropy of zero also. However, it does not have the symmetry property, due to the way that the conditional entropy is calculated in a particular order. (For example, with discrete RVs $H_c(X_1) = H(X_1)$, but if $X_2 = X_1$, then $H_c(X_2) = H(X_2|X_1) = 0$).

While the conditional entropy measure outlined above does not quite have the properties that we require, it is so close that we can use it as a basis to construct a symmetrical version that will.

5 Symmetrical conditional entropy

We saw in the previous section that conditional entropy relies on the following identity (in this case for a random vector):

$$H(X_1, \dots, X_n) = H(X_1) + H(X_2|X_1) + \dots + H(X_n|X_{n-1}, \dots, X_1). \quad (9)$$

However, there are $n!$ similar identities for the RVs X_1, \dots, X_n , depending on which order we choose them first. We can construct a symmetrical version of conditional entropy by using *all* of these identities added together, and collecting the terms with each X_i first. Rearranging, this will give us

$$\begin{aligned} n!H(X_1, \dots, X_n) &= (n-1)!(H(X_1) + \dots + H(X_n)) \\ &\quad + (n-2)!(H(X_1|X_2) + \dots + H(X_1|X_n)) \\ &\quad + H(X_2|X_1) + H(X_2|X_3) + \dots + H(X_2|X_n) \\ &\quad + H(X_3|X_1) + \dots + H(X_{n-1}|X_n) \\ &\quad + (n-3)!(H(X_1|X_2, X_3) + \dots + H(X_n|X_{n-1}, X_{n-2})) \\ &\quad + \dots \\ &\quad + (1)!(H(X_1|X_2, X_3, \dots, X_n) + \dots + H(X_n|X_{n-1}, \dots, X_1)) \end{aligned} \quad (10)$$

where we take the order of X_j s in these expressions to be significant. Collecting terms with each X_i first we can then express this as follows:

$$H(X_1, \dots, X_n) = \overline{H}_c(X_1) + \dots + \overline{H}_c(X_n) \quad (11)$$

where

$$\overline{H}_c(X_i) \triangleq \frac{1}{n} \left(H(X_i) + \sum_{k=1}^{n-1} H_c^k(X_i) \right) \quad (12)$$

and $H_c^k(X_i)$ is a ‘ k th-conditional-entropy’ term, which is the mean entropy in X_i given any k other X_j s, i.e.

$$H_c^k(X_i) = \binom{n-1}{k}^{-1} \sum_{j_1 > j_2 > \dots > j_k} H(X_i|X_{j_1}, \dots, X_{j_k}) \quad (13)$$

where all of the j_r s must be different to i .

Thus we have constructed a symmetrical conditional entropy measure $\overline{H}_c(X_i)$ specifically to fit our desired properties. In particular, it is additive, by verification of (11). It is symmetrical, due to the way that it has been constructed. Finally, it has a form of accountability property: in particular, we have

$$H(X_i)/m \leq \overline{H}_c(X_i) \leq H(X_i). \quad (14)$$

We therefore suggest the symmetrical conditional entropy $\overline{H}_c(X_i)$ for our entropy distribution measure.

With this measure now defined, we can go on to define the information distribution about as

$$\overline{I}_c(Y_i, X) = \overline{H}_c(Y_i) - \overline{H}_c(Y_i|X) \quad (15)$$

where $\overline{H}_c(Y_i|X)$ is defined in an analogous way to $\overline{H}_c(Y_i)$.

6 Example: two components

As an example of symmetrical conditional entropy being used for our entropy distribution measure, consider the case of a 2-D random vector $\mathbf{X} = (X_1, X_2)$. In this case we have

$$\begin{aligned} \overline{H}_c(X_1) &= 1/2 (H(X_1) + H_c^1(X_1)) \\ &= 1/2 (H(X_1) + H(X_1|X_2)) \end{aligned} \quad (16)$$

and similarly for $H(X_2)$.

Suppose that X_1 and X_2 have the same entropy, i.e. $H(X_1) = H(X_2)$. Then $\overline{H}_c(X_1) = (H(X_2) + H(X_1|X_2))/2 = H(X_1, X_2)/2$ which is simply the mean entropy.

Suppose instead that X_1 and X_2 are independent. Then $H(X_1|X_2) = H(X_1)$, so $\overline{H}_c(X_1) = H(X_1)$.

Note that in this 2-D case, since $I(X_1, X_2) = H(X_1) - H(X_1|X_2)$, we can write $\overline{H}_c(X_1) = H(X_1) - (1/2)I(X_1, X_2)$.

7 Alternative measures

The symmetrical entropy measure we have introduced above is not the only measure which satisfies the sort of properties we are interested in.

The measure above is one of the simplest. However, it does have properties which may make it awkward to use. For example, in the previous section, it was shown that in the two-variable case, the ‘shared’ entropy $I(X_1; X_2)$ between two variables is distributed equally to the variables. In fact, this generalizes to the n -variable case, where higher-order shared entropy terms are shared equally amongst contributing variables.

However, we may ask ourselves: what constitutes a ‘variable’ in this sense? If we treat a register that can take 16 values as one ‘variable’, we will get a different entropy distribution from the one we would get if we treat it as a vector of 4 binary ‘variables’. Is it possible to construct an entropy distribution measure which does not suffer from this problem?

One approach might be to consider that the critical feature of an entropy distribution measure is the way that it distributes the available information between two sets of variables when they are partitioned. We could therefore construct a entropy distribution function which depends on the marginal and joint entropy

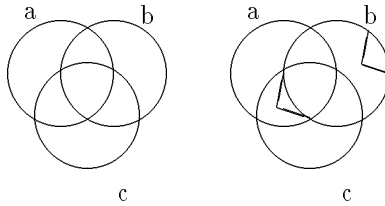


Figure 3: Venn-like diagram representing the entropy distributed to three sets of variables

measures, but not on the number of variables. However, further investigation reveals that any measure constructed in this way would be dependent on the order in which a set of information bearing channels were partitioned.

To see this, consider fig. 3, which is a Venn diagram showing how entropy is distributed to three sets of variables: \mathbf{X}_a , \mathbf{X}_b , \mathbf{X}_c . In the left hand figure, the area of circle ‘a’ represents the entropy $H(\mathbf{X}_a)$, the area in the intersection of circles ‘a’ and ‘b’ represents the mutual information $I(\mathbf{X}_a; \mathbf{X}_b) = H(\mathbf{X}_a) + H(\mathbf{X}_b) - H(\mathbf{X}_a, \mathbf{X}_b)$, and the small triangle represents the *triple information*

$$I(\mathbf{X}_a; \mathbf{X}_b; \mathbf{X}_c) = H(\mathbf{X}_a, \mathbf{X}_b, \mathbf{X}_c) + H(\mathbf{X}_a) + H(\mathbf{X}_b) + H(\mathbf{X}_c) - (H(\mathbf{X}_a, \mathbf{X}_b) + H(\mathbf{X}_b, \mathbf{X}_c) + H(\mathbf{X}_c, \mathbf{X}_a)) \quad (17)$$

which is calculated as we would expect from a set-like measure (except that triple information can be negative, which is the case if e.g. \mathbf{X}_a is the exclusive-or of \mathbf{X}_b and \mathbf{X}_c).

The right hand figure shows that it is possible to change the triple information without affecting any of the marginal or pair-wise joint entropy measures. Suppose we wish to partition the set of variables ‘a’, ‘c’ from ‘b’. If our information distribution function is only dependent on marginal and pair-wise measures, it must be independent of the triple information. However, we can show that if we then separate our set ‘a’+‘c’ into the separate sets ‘a’ and ‘c’ using the same measure, we will get a contradiction: the entropy distributed to ‘a’ will depend on the *order* that it was separated out: either (1) a+c from b, and then a from c; or (2) a from b+c.

For a better information distribution function, the most likely candidate appears to be a function of the information capacity of the set of variables as well as the entropy measures. This may avoid the partition ordering problem outlined above, but should allow us to produce a measure which is not dependent on the number of variables. Work is continuing in this direction.

8 Continuous domain

The extension of this entropy distribution concept to the continuous domain, to get an information density measure, is also not entirely straightforward. For example, it is not easy to define the equivalent of the k th conditional entropy H_c^k to allow for *all* other points on the real line. Current work suggests that a somewhat different approach may be needed to define information density over continuous domains.

For example, real neural fields have limited resolution, based on the density of neurons that they contain. Also, non-neural information-bearing fields (e.g. an image) may have limited resolution due to the point spread function of the lens used to focus the image. It may be that an acuity limit of some sort is

required to prevent the information density becoming infinite, in the similar way that measurement noise or inaccuracy prevents information rate becoming infinite for continuous-valued random variables.

9 Continuous example: gaussian random function

For the present, let us hypothesize that it is possible to define an information density function. In particular, we would like the concept of *information capacity density*, the maximum value of information density at a point in a neural field, to be well-defined.

Consider a normally distributed signal $Y(v)$ with mean zero and variance $S(v)$ at v . The real number v could be, for example, a position along a line in the spatial domain, or a frequency value in the frequency domain. Suppose that it is subject to additive noise of variance $N(v)$, and total power (variance) limit $P = \int S(v)$.

It is a well-known result from information theory, that the total information capacity of this system is maximized when $S(v) + N(v)$ is a constant [12]. For example, if $N(v) = N$ is constant with v (so-called *white* noise), then we should have $S(v) = S$. Thus the information capacity density $0.5 \log(S(v)/N(v))$ should also be uniform.

9.1 Frequency weighting solution

If v is a frequency f , then the obvious solution to this is the well-known *whitening filter* [12]. In this case, we simply apply a multiplier to the source of the signal $S(f)$ at each f so that the resulting signal is uniform over f . This is the solution achieved by a simple linear filter.

The disadvantage of this method is that it allocates equal information density to all frequencies. The solution is restricted to this since it is not a simple matter to change the frequencies used to transmit the information. However, if we are not working in the frequency domain, we have more flexibility in the solution we can adopt. In this case, it is possible to allocate more information density to certain areas.

9.2 Magnification solution

Suppose now that we have a system with input random function $X(u)$ defined over some continuous domain u , and an output random function $Y(v)$ similarly defined over some continuous v . Suppose also that we have a continuous map from an input range u to an output range $v(u)$. The output $Y(v)$ is determined from $X(u)$ ‘locally’ to $v(u)$, i.e. changes to $X(u)$ will cause the most significant effects in $Y(v)$ ‘close’ to $v(u)$.

In this case, we can get a non-uniform information capacity density by modifying the mapping $v(u)$. The information capacity density will be scaled in the forward direction by $1/(dv/du)$. Thus if we arrange to have uniform density at the output $Y(v)$, the input information density local to an input point u will be proportional to dv/du (fig. 4). The factor dv/du is the *magnification factor*.

If the information capacity density of the output field is uniform, this forward magnification factor dv/du is then simply proportional to the information density we wish to have in the input field at u .

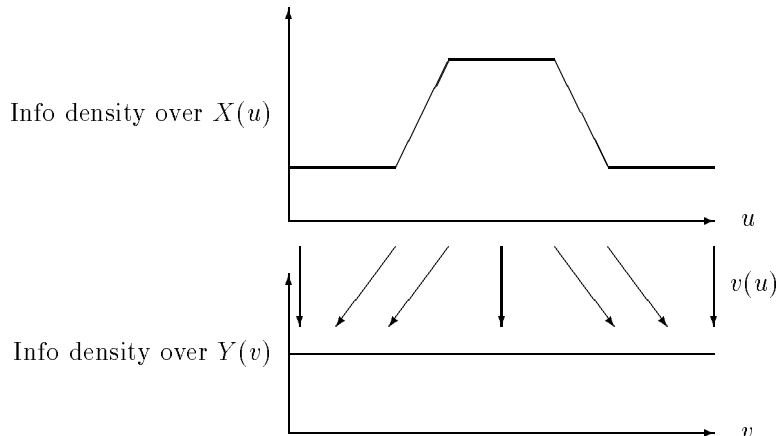


Figure 4: Magnification of input range over some regions of u , allowing a non-uniform information density over u , while we have a uniform information density over v .

10 Discussion

Provided that our hypothesized information density can in fact be defined, we now have an information-theoretic reason why magnification factors may be useful in any system that represents information over a continuous domain. We would anticipate a similar result from a domain such as a neural feature map, where information is represented by a large number of discrete elements, and with a total information capacity limit or power limit specified by the cost of representing information.

Going back to our biological examples, it offers a possible reason why the bat audio cortex, for example, should have the magnified representation around its radar frequency. The magnified representation gives the bat a much larger information density around that frequency than it does at others. Similarly in the human visual system, the magnification factor around the fovea from the retina to the visual cortex allows the information capacity density to be concentrated around the centre of the visual field.

The example of the fovea is also interesting since it is *mobile*: it is an area of high information capacity which can be moved across a scene at will. This may have implications for the operation of attention and visual scanning. It may be perhaps that a scanning fovea is used to perform concentration of information capacity density across both time and space.

We have not considered in this paper why maps should form locally, rather than have related information channels spread across the map. This is another avenue of current work, but we expect this to be related to the cost of computation over large distances in the cortex.

11 Conclusions

We have considered an alternative view to the investigation of feature maps, based on *information* density rather than *probability* density. The resulting type of map is conceptually somewhat different from the Kohonen self-organizing feature map (SOFM), and might be called an *information channel map* rather than a feature

map.

We discussed possible entropy distribution measures, which would have properties of summability, symmetry and accountability over a representation domain. These are not completely satisfactory at present, but they do point towards an information density measure which we hypothesize.

We supposed that the goal of an information channel map is to optimize information capacity. In this case, with a Gaussian signal, uniform output noise and a total power limit, the information density of the resulting map should be uniform. We therefore propose that the purpose of magnification factors found in biological systems is to match the non-uniform distribution of information density at the input to a sensory system. If a sensory system requires higher information density in one place than elsewhere (e.g. in a fovea), then the magnification factor will be higher there.

References

- [1] J. A. Anderson and G. E. Hinton. Models of information processing in the brain. In J. A. Anderson and G. E. Hinton, editors, *Parallel Models of Associative Memory*, pages 9–48. Lawrence Erlbaum Associates, Hillsdale, NJ, 1981.
- [2] J. J. Atick and A. N. Redlich. Towards a theory of early visual processing. *Neural Computation*, 2:308–320, 1990.
- [3] C. M. Bishop, M. Svensen, and C. K. I. Williams. GTM: A principled alternative to the self-organizing map. Technical Report NCRG/96/015, Neural Computing Research Group, University of Aston, UK, 1996.
- [4] S. Haykin. *Neural Networks: A Comprehensive Foundation*. Macmillan, New York, 1994.
- [5] T. Kohonen. Self-organized formation of topologically correct feature maps. *Biological Cybernetics*, 43:59–69, 1982.
- [6] R. Linsker. Self-organization in a perceptual network. *IEEE Computer*, 21(3):105–117, March 1988.
- [7] R. Linsker. How to generate ordered maps by maximising the mutual information between input and output signals. *Neural Computing*, 1:402–411, 1989.
- [8] S. P. Luttrell. Self-organization: A derivation from first principles of a class of learning algorithms. In *Proceedings of the 3rd International Joint Conference on Neural Networks, IJCNN'89*, volume II, pages 495–498, Washington, DC, June 18–22 1989. IEEE.
- [9] A. Papoulis. *Probability, Random Variables and Stochastic Processes*. McGraw-Hill, second edition, 1984.
- [10] M. D. Plumbley. Efficient information transfer and anti-Hebbian neural networks. *Neural Networks*, 6:823–833, 1993.
- [11] M. D. Plumbley. Lyapunov functions for convergence of principal component algorithms. *Neural Networks*, 8:11–23, 1995.
- [12] C. E. Shannon. Communication in the presence of noise. *Proceedings of the IRE*, 37:10–21, 1949.