April 2023 Issue

HITech

Technical magazine of CSE Department, Heritage Institute of Technology

In This Issue:

Foreword

Dr. Dinabandhu Bhandari, Professor, CSE

Before writing this foreword for the April 2023 issue of "HITech" the technical magazine of the Computer Science and Engineering, I tried to interact with the latest Artificial Intelligence (AI) tool "Chatbot" to get a write up that would help me to write this foreword. In my surprise, it produced one that I would not dare to put here because that may take away my individuality. In my opinion, Chatbot is a revolutionary addition in the hundreds of Chat applications available in the digital world over the last decade or so. In my opinion, it would be the Google, which has become synonymous to search engine, of chat applications. However, this is the beginning of the AI era that people are heavily trying to build over the last half of a century. As my teacher Prof. C. A Murthy used to say that the goal of the AI researchers is to mimic human intelligence. But people could be able to model only "δ" amount of only "ε" amount of human intelligence explored so far.

People are now busy with the development of different AI applications incorporating the human thinking process, perception and decision making ability. To accomplish this goal multidisciplinary research is inevitable and it is fascinating to observe the participation of experts of all subject areas in achieving this. The trend is clearly been reflected in "HITech" where most of the articles of our students are related to AI and their applications to different areas. This also demonstrates our primary goal to encourage students in the latest developments of computer science.

It gives me immense pleasure to write this foreword for "HITech" magazine. I sincerely thank the editorial board to give me this opportunity. I appreciate all the students and staff members of CSE department for their fruitful contribution in the magazine.

Density of Points in the Euclidean Space of Different Dimensions

By, Prof. Nilanjana G. Basu, Assistant Professor, CSE

The act of characterizing and measuring different attributes of primitive shapes is of paramount importance in the subject of computational and discrete geometry. A very rich collection of work can in fact be found in this domain, which is predominantly focused on Euclidean space. The term ‘density of points’ provides a notion of the relative concentration or rarefaction of a collection of points within a given shape or region and it has plenty of applications in different branches of physical science. For a set S of scattered points, a very frequently asked question is whether a cluster $C \in S$ is more concentrated or less saturated with respect toS. Discrepancy theory can be described as the study of inevitable irregularities in the distribution of a set of points. Just as Ramsey theory explains the impossibility of total disorder, discrepancy theory studies the deviations from total uniformity.

In 2D, ‘Density of points’ of an unweighted set of points is expressed as the number of points per unit area. In case of a weighted set of points, it is the sum of the weights divided by the area of that region. A very counter-intuitive result was found by Majumder and Bhattacharya[1] , which said that an axes-parallel region of maximum (minimum) density would always contain only two (one) of the points from the given set of points. They also proposed an efficient algorithm [1] to identify the rectangular region of maximum density in R^2 when the points are of uniform weight.

Thermal analysis of VLSI chips provides deeper insights into their behavior under different temperature profiles and enables faster design exploration. If the thermal contribution of the heat sources of a particular region cross the saturation level, the thermal balance will be difficult to achieve, resulting in performance degradation of the chips. If the highly active heat sources can be identified and moved closer to a region where heat sinks are present or less number of heat sources are located, the chips could be cooled down. This had motivated the original work[1] on finding the maximum density axes-parallel rectangle in R^2 for weighted as well as unweighted points. However, the concept of density of points and its analysis did not remain limited to 2D only. Later, geometers studied the same problem for weighted point sets in R^2 and in R^3, where the weight of a point may represent the relative strength of any physical parameter.

The search for density extrema is quite challenging not only in continuous space, but also in digital space, when the set S of discrete points with uniform weight are present at the crossing points of a uniform rectilinear(cuboidal) grid. Consider discs moving on an integer plane, with changing center and varying radius, i.e., center can be aligned with a grid point or may be any real point and radius can be either an integral multiple of the grid unit or can be relaxed to be any real number. In this geometric setup, locating the density extrema for the maximum- and minimum-density circular region in 2D and in 3D is a new challenge to solve. This problem might be extended to n-dimension as well.

According to l_2 vector norm, for an l_2 ball, the square root of the sum of the squares of the component vectors along the axes is constant. Say in 2D, (x, y) is the coordinate of any point on the circumference of a disc. As $\sqrt{(|x|^2 + |y|^2)}$ is the radius of the disc, $\sqrt{x^2 + y^2}$ = constant specifies a disc in l_2 norm. The l_2 norm of a point is basically the Euclidean distance of the point from the origin.

As discussed, a disc is an l_2-ball. Similarly, a square is an l_∞-ball. According to l_∞ norm, distance of a point x from the origin, $|x|_\infty = \max_i |x_i|$. The l_∞ norm of a point in R^2 is just the maximum of the absolute values of the two coordinates. Say, (x, y) is the coordinate of any point on a square. As $\max(|x|, |y|)$ is always constant for any point on a square centered at origin, a square is basically an l_∞-ball. An interesting problem will be to identify the density extremas of square, cube and hypercubes in different dimensions having integral and real length.

Another commonly used vector norm is the l_1 norm. According to l_1 norm, the distance of a point (x, y) in 2D from the origin is $|x| + |y|$, i.e., the sum of the magnitudes of the component vectors along the axes. In other words, the l_1 norm of a point in R^2 is just the manhattan distance of the point from the origin (0,0). Say, (x, y) is the coordinate of any point on a diamond (rhombus) centered at origin. As $|x| + |y|$ is always constant for any point on a diamond, it is an l_1-ball. Locating the highest and the lowest density diamond when the length of a diamond can be either odd or even integer or any real number, is another interesting problem to study.

Among mathematicians the shape hexagon is widely popular for its inherent properties which makes it unique and naturally occurring. The hexagonal shape fills a plane in the best manner with equal size units and leaves no space wasted. For example, the common snowflake patterns, the north pole of Saturn, south pole of Jupiter formed due to super storm, bee honeycombs and more interestingly, the Giant's Causeway in Northern Ireland are all of hexagonal shape. Also, hexagonal graphite is the thermodynamically stable form of graphite. Naturally, the problem of identifying the location of the maximum- and minimum-density hexagon would be a pertinent topic of research.

The hexagonal and the triangular grids are duals of each other. Be it calculating the eigen values of equilateral triangles, random distances associated with them, identifying the maximum number of equilateral triangles determined by a set of n points in any dimension or analyzing three-plane and n-plane triangular grids, the equilateral triangle is always a topic of interest among mathematicians. Downward equilateral triangles are the one with one of their sides parallel to x-axis and the opposite corner is below that side. For a set of points in general position, the size of a matching for the class of graphs of downward equilateral triangles has been deducted. So, identifying the density extremas of equilateral triangles is another very important area to explore.

We conclude that the estimation of density extrema for a set of points is a rich problem with a lot of variations. There is still ample scope for finding new problems in the domain of the density of points, a natural extension being the consideration of the basic shapes in arbitrary orientation. However, some of them may be found to be prohibitively hard to address.

References

[1] S. Majumder and B. B. Bhattacharya. On the density and discrepancy of a 2d point set with applications to thermal analysis of vlsi chips. Inf. Process. Lett., 107(5):177–182, 2008.

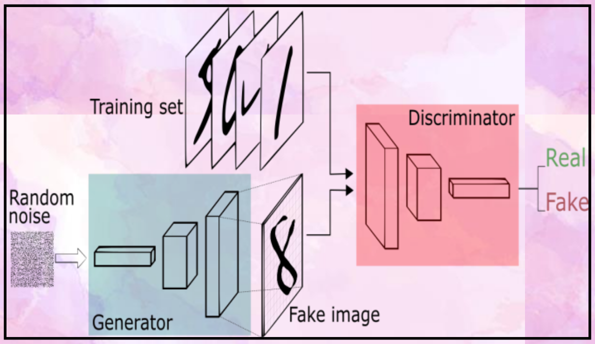
Applications of Generative Adversarial Networks(GANs) in Medical Imaging

By Suchandra Chakraborty, Srinjoy Bhuiya, Subhopriyo Sadhukhan, Apurva Daga, 4th Year, CSE

Modern medicine is not complete without medical imaging, which enables doctors to diagnose and cure a wide range of illnesses accurately. The use of Generative Adversarial Networks (GANs) in medical imaging has revolutionized the manner in which doctors analyze medical images. GANs have the potential to generate synthetic images that can be used in several medical applications such as data augmentation, image de-noising, image segmentation, and illness diagnostics.

Basic Architecture of GANs

Generative Adversarial Networks (GANs) are a popular and powerful class of deep-learning models used to generate synthetic data. GANs consist of two neural networks: a generator and a discriminator. The generator network takes a random noise vector as input and produces synthetic data, whereas the discriminator network takes both real and synthetic data and distinguishes between them. As the training progresses, the generator network gradually learns to generate synthetic data that are increasingly realistic and challenging for the discriminator network to differentiate from real data.



The Architecture of Vanilla GANs.

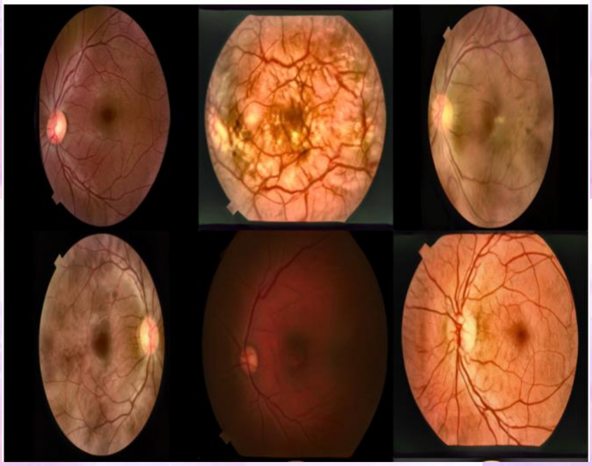
Variations of GANs used in medical imaging:

The following are some variations of GANs that have been employed in medical imaging:

(1) Pix2Pix, a type of conditional GANs, uses a paired training dataset to learn the mapping between an input image and an output image. In medical imaging, Pix2Pix has been used for tasks such as image segmentation, where the goal is to generate a binary mask that highlights the regions of interest in the input image. (2) CycleGAN can learn the mapping between two image domains without requiring paired training data by training two generators and two discriminators, where the generators learn to translate images from one domain to another and the discriminators try to distinguish between the generated and real images. CycleGAN has been utilized in medical imaging for applications such as cross-modality synthesis and image-to-image translation. (3) StyleGAN can generate high-quality images with fine-grained control over the style and content of generated images. It uses a novel generator architecture that incorporates style vectors to control the image-generation process. In medical imaging, StyleGAN has been used to generate realistic medical images for training machine-learning models as well as for data augmentation. (4) WGAN-GP uses gradient penalty to improve the stability and training of the discriminator. It has been demonstrated to increase GAN training stability and yield more realistic images. WGAN-GP has been utilized in medical imaging for tasks like picture synthesis and data augmentation.
These are only a handful of GAN variations that have been applied to medical imaging. Each variant may be customized for particular activities and applications and has advantages and disadvantages.

Using GANs in Medical Image Classification and Segmentation:
GANs have shown remarkable promise in medical image classification and segmentation. Medical images frequently contain noise and anomalies, which makes successful classification and

segmentation difficult. GANs can help boost the classification and segmentation job accuracy by creating high-quality pictures and reducing noise. They can also be used to perform image segmentation by constructing a mask of the target region in the image. For example, a GAN may be taught to segment lung areas in CT images or to detect tumors on MRI scans. The generated masks can subsequently be used to classify the images correctly. GANs can also be used to categorize medical images by producing additional images that can aid the identification of specific features in an image. For instance, a GAN can generate images that highlight specific aspects of a tumor or organ, making the image simpler to categorize. GANs can also be used to solve the data scarcity challenges in medical imaging. Medical imaging datasets are frequently limited, making it challenging to successfully train machine-learning models. GANs can be used to generate synthetic medical pictures that can then be mixed with real-world medical images to form a larger and more diversified dataset. This method has the potential to improve the performance of machine-learning models, resulting in more accurate and robust diagnoses.



Images of the Retina Fundus that were generated using StyleGAN2 ADA

In conclusion, the application of GANs in medical imaging has shown to be an effective tool for producing synthetic images, enhancing image classification and segmentation, and addressing data scarcity issues.With further research and development, GANs have the potential to transform medical imaging and contribute significantly to the accuracy and robustness of diagnoses, ultimately leading to better diagnosis of diseases.

Some Reflections on Software Engineering Education

By Dr. Subhajit Dutta, Associate Professor, CSE

Education is an interesting topic. Quiet creatures can be raised to strident disquiet when talking about education, especially someone else's. To play it safe, I have qualified "education" in the title with "software engineering" and qualified "software engineering education" with "some reflections". This is neither a complete nor a consistent commentary on education in the abstract.

Krutchen has conjectured "that the half-life of software engineering ideas is roughly five years". That is, five years from a given date, 50% of what constitutes the current state of the art "... will have been forgotten or seriously marginalized – not really worth teaching an undergraduate software engineering student" (The biological half-life of software engineering ideas. *IEEE Software*, 25(5):10–11, 2008). This seems quite plausible to those associated with software engineering for more than five years. This also raises serious questions on what *should be* taught to the student of software engineering. If what is *cool* today will be cold in just five years, which hot topics should we teach now?

This is a problem common to any fast evolving field. Hamming's remark in the general context of science and engineering education is apt for software engineering too: "... teachers should prepare the students for the student's future, not the teacher's past" (*The Art of Doing Science and Engineering: Learning to Learn*. CRC Press, 1997). That is easier said than done; certainly for those of us not endowed with Hamming's wisdom and experience. But I believe it can serve as an enduring vision for the way software engineering is taught and learnt. We have full knowledge of our past, and none about the student's future. Yet, we must strive to impart an understanding of the problems today's students will be called upon to solve tomorrow.

The progress of any discipline can be seen as a window moving from left to right across a continuum of change. The frame of the window represents the present. On the left we have concepts and techniques that worked before but have since been found to be inadequate. On the right we have exploratory results that look promising but are yet to be tried beyond controlled conditions. In the middle, within the window's frame, lie today's state-of-the-art, nourished by insights from the left, and feeding experience to the right. The window may move relatively fast for a field like software engineering, but this continuum is common across fields. A key aspect of preparing students for the future is to sensitize them to the *moving* nature of the window. In addition, we need to culture an appreciation of what worked before, what works now, and what is likely to work in the future.

On my good days I can convince myself I am not that old (yet); still a lot has changed from the time I was an undergraduate student. Then, the snazziest gadget we had was a wrist-watch (which only gave us time); Android sounded like a humanoid robot from a science fiction, the Internet was something like the Aurora Borealis, we knew it existed, but nearly no one had seen it. The biggest difference between then and now is the permeance of screens in our lives, all connected to the World Wide Web.

Education is a curious chemistry of information and insight. Without the Web, teachers and books were the only source of information, as well as insight. But with the Web, information is just clicks or taps away. But insight is still hard to glean. Insight comes from distilling information in a particular context, and depends heavily on experience and intuition of the distiller. The challenge for today's teachers is to move from being purveyors of information to becoming kindlers of insight.

And, what is the half-life of that challenge?

Text guided Image Generation Overview

By, Somnath Dey, Sachin Kumar, Shubham Kashyap, Ayanika Chaudhury, 4th Year, CSE

WHAT IS IT?

Text Guided Image Generation is a process in computer vision and natural language processing where the goal is to generate an image that corresponds to a given prompt (textual description). Text to image generating deep learning models generally combine a language model, which transforms the input prompt into a vector representation, and a generative image model, which produces an image conditioned on that representation. The most effective models have generally been trained on massive datasets of images paired with text captions, E.g., COCO by Microsoft.

WHEN DID IT START?

The first modern text-to-image model, **alignDRAW**, was introduced in 2015 by researchers from the University of Toronto. Images generated by **alignDRAW** were blurry and not photorealistic, but the model was able to generalize to objects not represented in the training data and appropriately handled novel prompts such as "a stop sign is flying in blue skies", showing that it was not just merely memorizing data from the training dataset. In 2016, Reed, Akata, Yan et al. was the first to use generative adversarial networks (GANs) for text-to-image generation. In later models VQGAN+CLIP, XMC-GAN, and GauGAN2 were included. Later **Diffusion** and **Stable diffusion** models were introduced in this field of deep learning.

HOW DOES IT WORK?

Generative Adversarial Networks (GANs): Generative adversarial networks (GANs) consist of a discriminator which tries to distinguish real training data from synthetic images, and a generator which tries to fool the discriminator. The prompt is first transformed into a text embedding, concatenated with a noise vector and then given as input to the Generator which helps to generate images corresponding to the input description instead of generating random images. In the Discriminator a pair of image and text embeddings are sent as input. Output signals are either 0 or 1. Along with identifying whether the given image is real or fake, it also predicts how much the given image and prompt corresponds to each other. DCGAN, CycleGAN, StyleGAN, PixelRNN are popularly used GAN models for text guided Image generation purposes.

Diffusion Models: Diffusion Models define a Markov chain of diffusion steps by destroying training data (forward diffusion) through the successive addition of Gaussian noise, and then learning to recover (reverse diffusion) the data by reversing this noising process. After training, we can generate images by passing randomly sampled noise through the learned denoising process. The latent variable in diffusion models has high dimensionality (same as the original data).

Stable Diffusion Models: Stable Diffusion is a latent diffusion model which was first introduced in August 2022 by Stability AI. Instead of operating in the high-dimensional image space, it first compresses the image into the latent space. The latent space is 48 times smaller, which is why it’s a lot faster.

It consists of three models and a scheduler. These are:

Clip Text Encoder: It encodes the text prompt into text embeddings which is a tensor of 77x768 dimension.

Variational Autoencoder (VAE) encoder / decoder: The VAE encoder is used to convert the image (512x512px) with 3 channels into a low dimensional latent (1x4x64x64) representation, which serves as the input to the U-Net model. The decoder, conversely, transforms the latent representation back into an image.

Text conditioned latent U-Net: The U-Net has an encoder part and a decoder part both composed of ResNet blocks of measurely Conv2ds. By reversing the noising process, it predicts the noise to be removed from the latents based on the conditioned text embedding provided as input.

Schedulers: These are used to divide the complete diffusion process into n (given as input) steps and removes the noise from latent using the predicted noise by U-Net at each step.

Stable Diffusion v2 was released in November 2022 which includes robust text-to-image models trained using new text encoder OpenClip which greatly improves the image quality compared to v1 and it can also generate images with default resolutions both 768x768 pixels and 512x512 pixels. Stable Diffusion v2 also includes an upscaler diffusion model that enhances image resolution by a factor of 4.

DALL-E and DALL-E 2 by OpenAI, Google Imagen, Midjourney by David Holz are other examples of diffusion model projects for generating images guided by text prompts.

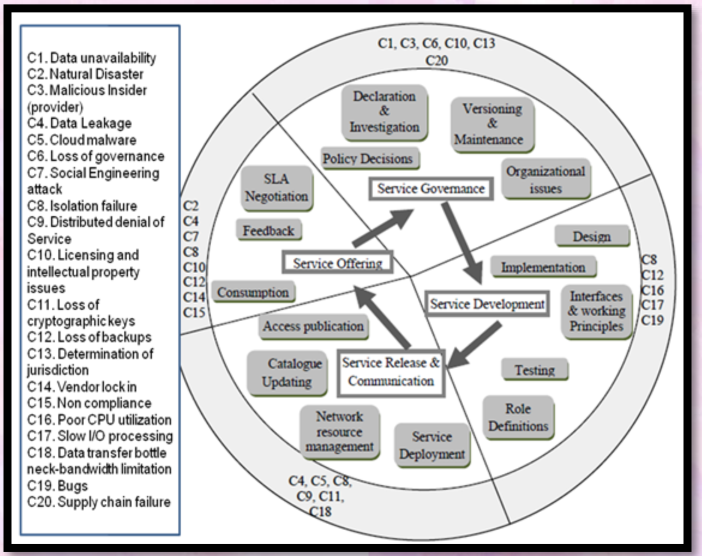
WHERE CAN IT BE USED?

Text guided image generation is only in its early stage. It holds enormous possibilities in the world of Artificial Intelligence and Machine Learning. It can be used for creating visual representation of the scenes in stories, creating compelling and eye-catching visual ads (recently coca cola made one), generating realistic images of clothing and fashion accessories, etc

Cloud Service Insurance - Quality Assurance to Consumers

By Dr. Adrija Bhattacharya, Assistant Professor, CSE

Service cost is most sensitive to its performance. Mostly the service performance is considered to be good if the service can resolve any probable failure and continue to provide seamless offering of the subscribed service. The service failure can occur due to occurrence of any risks that a service is susceptible to throughout its lifetime (attached figure). But it is also true that every time a service does not fail when a particular risk (cause) occurs. Because often service oriented system contains risk resolution mechanisms. This mechanism helps to mitigate the issues raised in ongoing service executions at runtime. Recovery from the risks on the fly or in minimum time is referred as service recoverability. In spite of existing recovery capabilities, a service may fail with the occurrence of certain risks. The quality performance of a system degrades with the failure of any component service in terms of committed QoS. This could be a serious concern for a critical service (for example in healthcare domain) since it deprives the stakeholder and eventually exposes to potential harms.



In a subscription based pricing model service failure costs high for the service providers in terms of reputation. In such a competitive market only the service provision is not enough to satisfy user but some guaranteed QoS offerings would highly repute a provider. The idea of service insurance is developed to safeguard consumer's side with guaranteed QoS. It includes a cost component into any existing pricing models. The extra cost should be paid for the guaranteed QoS. i.e. while allocating resources the extra paying customers should get privilege over general customers. This idea would be implemented by means of Cloud Service Brokers This was motivated when the service lifecycle shows similarity with human lifecycle and the vulnerability in service life is affecting the service life (failure), performance dynamically. As a consequence the price is rising and the consumer dissatisfaction with quality continues. This is very similar with the life loss or physical/clinical damage caused in human life. Thus the idea of insurance is used to safeguard service value for money for consumers and the mechanism helps to decompose the failure into causes involving several underlined KPIs. The similarities between service and Human insurances are discussed as follows-

- Human Insurance (HI) provides cost against life loss. Service Insurance (SI) provides cost repayment against failure. Alternately it forces provider to maintain guaranteed quality of insured services to consumer.
- Life expectancy of a human is based on several factors (family history, demography, social status, etc.) is calculated. SI includes service related dynamic factor while calculating the service life expectancy
- Paying some premium the HI can be bought and that will safeguard human life as well as illnesses. SI also comes in terms of premium. Getting insured will help consumer service subscription at a guaranteed level of QoSs. While failing to provide intended QoS, claims can be done.

Depending on e detailed risk modeling and failure estimation the above said idea can be initially conceptualized. However, the risk causes identified have high impact on business management. This will pioneer an avenue toward newer business systems. This risk model may be extended by mitigating failures from interdependent causes. Similarly, application of actuarial mathematics in the domain, a comparative study among several actuarial and risk modeling, and forecasting methods are the areas for future explore.

Organ Engineering

By, Dr. Anindita Kundu, Assistant Professor, CSE

Last year, it came into news that, surgeons at the University of Maryland had transplanted a heart from a genetically engineered pig into a human. The 57-year-old man named David Bennett managed to live with a pig heart beating inside his chest for about two months. This first-of-its-kind surgery was hailed as a major step forward in the decades-long effort to successfully transplant animal organs into humans.

According to the United Network for Organ Sharing, which oversees a network for organ transplants in the United States, there were more than 3,800 heart transplants in the country last year — a record number. More than 100,000 people in the U.S. alone are currently waiting for an organ transplant, and over 6,000 people die each year before getting one, according to the federal government's organdonor.gov website.

Animal organs are a potential solution for now. But it's not easy to overcome the human body's natural revolt against them. For example, sugars on the surface of pig tissue can send our immune system into attack mode. Drugs can help mute the response, but it's not enough. So generic engineering comes into the picture. Scientists are working to modify pig's genes by removing those sugar molecules and adding other genes to make the 'pigs' seem more 'human-like'.

Research has now resulted in the creation of new genetically modified animal species whose organs are more compatible with human bodies. Though Bennett could not survive long, as a virus was found in the transplanted organ, his doctors claim the pig heart he received never developed classic signs of organ rejection. Hence, they're planning studies with more patients.

Martine Rothblatt, a successful satellite entrepreneur, started a biotechnology company, United Therapeutics after her daughter Jenesis was diagnosed with a fatal lung disease. Special Drugs are now keeping many patients like Jenesis alive but she might eventually need a lung transplant. Rothblatt therefore had set out to solve that problem too, using technology to create what she calls an "unlimited supply of transplantable organs."

The entrepreneur explained her plans with the help of an architect's rendering of an organ farm set on a lush green lawn, its tube-like sections connected whimsically in a snowflake pattern. Solar panels dotted the roofs, and there were landing pads for electric drones. The structure would house a herd of a thousand genetically modified pigs, living in strict germ-free conditions. There would be a surgical theater and veterinarians to put the pigs to sleep before cutting out their hearts, kidneys, and lungs. These lifesaving organs—designed to be compatible with human bodies—would be loaded into electric copters and whisked to transplant centers.

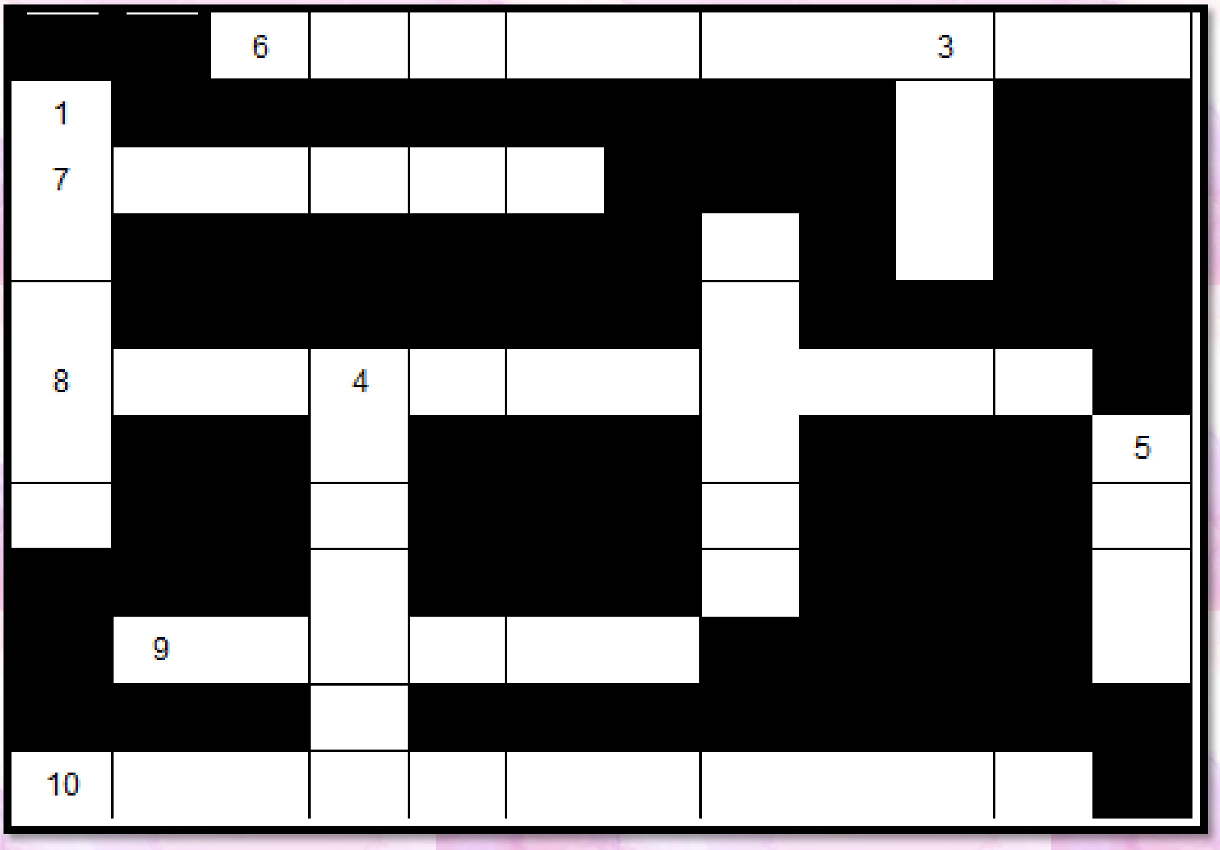
Obviously the idea is opposed by many human rights groups like PETA and there are issues of religion too. Although Jewish law forbids Jews from raising or eating pigs, receiving a pig heart is "not in any way a violation of the Jewish dietary laws", says Dr Moshe Freedman, a senior London rabbi who sits on the UK Health Department's Moral and Ethical Advisory Group (MEAG).

In the future, organ engineering might not involve animals at all. Researchers are in the early stages of exploring how to engineer complex tissue from the ground up. Some are 3D-printing scaffolds in the shape of lungs. Others are cultivating blob-like "organoids" from stem cells to imitate specific organs. In the long term, researchers hope to grow custom organs in factories.

Whether they're grown in animals or built inside manufacturing plants, an unlimited supply of organs could make transplantation more common, and give far more people access to replacement parts.

Crossword

Collected by, Prof. Somenath Sengupta, Assistant Professor, CSE



DOWN

1. The class members declared as _____ can be accessed only by the member functions inside the class
2. A ______is a collection of statements that perform some specific task and return the result to the caller.
3. The _____ pointer is passed as a hidden argument to all non-static member function calls and is available as a local variable within the body of all non-static functions.
4. In _____ polymorphism, the compiler resolves the object at run time, and then it decides which function call should be associated with that object.
5. The class _____ is surrounded by braces, { }.

ACROSS

6. _____ is an instance member function which is invoked automatically whenever an object is going to be destroyed
7. The _____ statement returns the flow of the execution to the function from where it is called.
8. An _____ is a collection, or the gathering of things together.
9. _____ function can be granted special access to private and protected members of a class in C++.
10. The capability of a class to derive properties and characteristics from another class is called _____ .

Solution

		D	E	S	T	R	U	C	T	O	R
P									H		
R	E	T	U	R	N				I		
I							M		S		
V							E				
A	G	G	R	E	G	A	T	I	O	N	
T			U				H				B
			N				O				O
			T				D				D
	F	R	I	E	N	D					Y
			M								
I	N	H	E	R	I	T	A	N	C	E	