shape algebra

shape algebra is a fascinating area of mathematics that combines the concepts of geometry and algebra to explore the properties and relationships of shapes through mathematical expressions and equations. This field is crucial for various applications, including computer graphics, engineering, and even architectural design. In this article, we will delve deep into the fundamentals of shape algebra, discussing its principles, methods, and real-world applications. We will also provide insights into the various tools and techniques used in shape algebra, as well as its significance in educational settings. By the end, readers will have a comprehensive understanding of shape algebra and its importance in both theoretical and practical contexts.

Understanding the Basics of Shape Algebra

Shape algebra is a mathematical framework that allows for the manipulation and analysis of geometric shapes using algebraic expressions. At its core, shape algebra combines the properties of shapes with

algebraic operations to form equations that describe these shapes. This dual approach enables mathematicians and scientists to model complex systems and phenomena that involve spatial relationships.

In shape algebra, shapes can be represented in various forms, such as points, lines, polygons, and curves. Each shape possesses unique properties, such as area, perimeter, and angles, which can be expressed in algebraic terms. For instance, the area of a rectangle can be expressed as the product of its length and width, while the perimeter can be calculated by adding together the lengths of all four sides. This mathematical representation is fundamental in analyzing the relationships between different shapes.

Key Concepts in Shape Algebra

To effectively engage with shape algebra, one must understand several key concepts that underpin this field. These concepts include transformation, congruence, similarity, and the use of coordinates.

Transformation

Transformation in shape algebra refers to the process of changing the position, size, or orientation of a shape without altering its fundamental properties. There are several types of transformations, including:

- **Translation:** Moving a shape from one location to another without changing its size or orientation.
- **Rotation:** Turning a shape around a fixed point at a certain angle.
- **Reflection:** Flipping a shape over a line to create a mirror image.
- **Scaling:** Increasing or decreasing the size of a shape while maintaining its proportions.

Congruence

Congruence is a critical concept in shape algebra that describes when two shapes are identical in form and size, allowing them to coincide perfectly when superimposed. Congruent shapes share the same dimensions and angles, which is essential for solving geometric problems.

Similarity

Similarity, on the other hand, refers to shapes that have the same form but differ in size. Similar shapes maintain proportional dimensions and angles, making them useful in various applications, such as scale modeling and architectural design.

Use of Coordinates

Shape algebra often employs a coordinate system to represent geometric figures. The Cartesian coordinate system, which uses pairs of numerical coordinates to define points in a plane, is particularly common. This approach enables algebraic manipulation of shapes through equations, facilitating analysis and problem-solving.

Applications of Shape Algebra

Shape algebra has widespread applications across several fields, showcasing its versatility and importance. Some of the key areas where shape algebra is utilized include:

Computer Graphics

In computer graphics, shape algebra is essential for rendering images and animations. By using mathematical models to represent shapes, programmers can create realistic simulations and visual effects in video games and movies. Shape algebra allows for the manipulation of geometric objects,

enabling transformations that enhance visual storytelling.

Engineering

Engineers frequently apply shape algebra in design and analysis processes. For instance, in structural engineering, understanding the properties of shapes is crucial for ensuring stability and strength in buildings and bridges. Shape algebra aids in optimizing designs to achieve both functionality and aesthetic appeal.

Architecture

Architects rely on shape algebra to conceptualize and visualize their designs. By utilizing mathematical principles to understand the relationships between different shapes, architects can create innovative structures that are both functional and visually striking. Shape algebra also facilitates the analysis of light, space, and materials in architectural design.

Tools and Techniques in Shape Algebra

A variety of tools and techniques are employed in shape algebra to facilitate the analysis and manipulation of geometric shapes. These tools range from software applications to theoretical approaches.

Software Tools

Modern technology has greatly enhanced the capabilities of shape algebra through specialized software. Some popular tools include:

- **Geometric Modeling Software:** Applications like AutoCAD and Rhino allow users to create, manipulate, and analyze geometric shapes with precision.

- **Computer-Aided Design (CAD):** CAD software is widely used in engineering and architecture for designing and drafting technical drawings.

- **Mathematical Software:** Programs such as MATLAB and Mathematica provide powerful computational tools for performing algebraic manipulations on geometric shapes.

Theoretical Techniques

In addition to software tools, theoretical techniques play a significant role in shape algebra. Techniques such as algebraic geometry and topology provide a deeper understanding of the properties and relationships of shapes. Algebraic geometry, in particular, explores the connection between algebraic equations and geometric figures, offering insights into complex spatial relationships.

The Role of Shape Algebra in Education

Shape algebra is an integral part of mathematics education, helping students develop critical thinking and problem-solving skills. By introducing students to the principles of shape algebra, educators can foster a deeper understanding of geometry and its applications.

In the classroom, shape algebra can be taught through interactive activities that encourage exploration and experimentation. Students can engage in hands-on projects that involve creating and manipulating shapes, allowing them to visualize mathematical concepts in a tangible way. This experiential learning approach enhances student comprehension and retention of geometric principles.

Future Trends in Shape Algebra

The field of shape algebra continues to evolve, driven by advancements in technology and an increasing demand for innovative applications. Future trends may include:

- **Integration with Artificial Intelligence:** As AI technology progresses, shape algebra may play a crucial role in developing intelligent systems that can analyze and interpret shapes in real time.

- **Enhanced Visualization Techniques:** New visualization tools could provide more intuitive and interactive ways to explore geometric shapes, making shape algebra more accessible to a broader audience.

- **Interdisciplinary Applications:** Shape algebra is likely to find applications in emerging fields such as robotics, virtual reality, and bioinformatics, demonstrating its versatility and relevance in solving complex problems.

Overall, shape algebra is a dynamic and essential field of study that bridges the gap between geometry and algebra. Its applications are vast, and its principles are foundational to understanding the spatial relationships that govern our world.

Q: What is shape algebra?

A: Shape algebra is a mathematical framework that combines geometry and algebra to analyze and manipulate geometric shapes through algebraic expressions and equations.

Q: How is shape algebra used in computer graphics?

A: In computer graphics, shape algebra is used to represent and manipulate geometric objects, enabling the creation of realistic images and animations through transformations and modeling techniques.

Q: What are the key transformations in shape algebra?

A: The key transformations in shape algebra include translation, rotation, reflection, and scaling, which change the position, size, or orientation of shapes without altering their fundamental properties.

Q: How does shape algebra benefit engineering and architecture?

A: Shape algebra benefits engineering and architecture by providing mathematical models that help in optimizing designs for stability, functionality, and aesthetic appeal, as well as facilitating analysis of spatial relationships.

Q: What tools are commonly used in shape algebra?

A: Common tools used in shape algebra include geometric modeling software like AutoCAD, computer-aided design (CAD) applications, and mathematical software such as MATLAB and Mathematica.

Q: What role does shape algebra play in education?

A: Shape algebra plays a significant role in education by helping students develop critical thinking and problem-solving skills, as well as fostering a deeper understanding of geometry through interactive learning activities.

Q: What are some future trends in shape algebra?

A: Future trends in shape algebra may include integration with artificial intelligence, enhanced visualization techniques, and interdisciplinary applications in fields such as robotics and virtual reality.

Q: Can shape algebra be applied to real-world problems?

A: Yes, shape algebra can be applied to real-world problems in various fields, including engineering, architecture, computer graphics, and more, providing solutions to complex spatial relationships and design challenges.

Q: What is the significance of congruence and similarity in shape algebra?

A: Congruence and similarity are significant in shape algebra as they describe relationships between shapes; congruent shapes are identical in size and form, while similar shapes maintain proportional dimensions, which is vital for analysis and design.

Shape Algebra

shape algebra: Shape George Stiny, 2006-04-07 How design is calculating with shapes: formal details and design applications.

shape algebra: Shape Theory J. M. Cordier, T. Porter, 2013-12-01 This in-depth treatment uses shape theory as a case study to illustrate situations common to many areas of mathematics, including the use of archetypal models as a basis for systems of approximations. It offers students a unified and consolidated presentation of extensive research from category theory, shape theory, and the study of topological algebras. A short introduction to geometric shape explains specifics of the construction of the shape category and relates it to an abstract definition of shape theory. Upon returning to the geometric base, the text considers simplical complexes and numerable covers, in addition to Morita's form of shape theory. Subsequent chapters explore Bénabou's theory of distributors, the theory of exact squares, Kan extensions, the notion of a stable object, and stability in an Abelian context. The text concludes with a brief description of derived functors of the limit functor theory—the concept that leads to movability and strong movability of systems—and illustrations of the equivalence of strong movability and stability in many contexts.

shape algebra: Foundations of Algebraic Specification and Formal Software Development Donald Sannella, Andrzej Tarlecki, 2012-01-05 This book provides foundations for software specification and formal software development from the perspective of work on algebraic

specification, concentrating on developing basic concepts and studying their fundamental properties. These foundations are built on a solid mathematical basis, using elements of universal algebra, category theory and logic, and this mathematical toolbox provides a convenient language for precisely formulating the concepts involved in software specification and development. Once formally defined, these notions become subject to mathematical investigation, and this interplay between mathematics and software engineering yields results that are mathematically interesting, conceptually revealing, and practically useful. The theory presented by the authors has its origins in work on algebraic specifications that started in the early 1970s, and their treatment is comprehensive. This book contains five kinds of material: the requisite mathematical foundations; traditional algebraic specifications; elements of the theory of institutions; formal specification and development; and proof methods. While the book is self-contained, mathematical maturity and familiarity with the problems of software engineering is required; and in the examples that directly relate to programming, the authors assume acquaintance with the concepts of functional programming. The book will be of value to researchers and advanced graduate students in the areas of programming and theoretical computer science.

shape algebra: *Accessible Algebra* Anne Collins, Steven Benson, 2023-10-10 Accessible Algebra: 30 Modules to Promote Algebraic Reasoning, Grades 7-10 is for any pre-algebra or algebra teacher who wants to provide a rich and fulfilling experience for students as they develop new ways of thinking through and about algebra.' The book includes 30 lessons that identify a focal domain and standard in algebra, then lays out the common misconceptions and challenges students may face as they work to investigate and understand problems.' Authors Anne Collins and Steven Benson conferred with students in real classrooms as the students explained what problem-solving strategies they were using or worked to ask the right questions that would lead them to a deeper understanding of algebra. Each scenario represents actual instances of an algebra classroom that demonstrate effective teaching methods, real-life student questions, and conversations about the problems at hand. 'Accessible Algebra' works for students at every level. In each lesson, there are sections on how to support struggling students, as well as ways to challenge students who may need more in-depth work. There are also numerous additional resources, including research articles and classroom vignettes.

shape algebra: The Shape of Algebra in the Mirrors of Mathematics Gabriel Katz, Vladimir Nodelman, 2012 The Shape of Algebra is the authors' attempt to share their mathematical experiences with readers who have more than a passing interest in mathematics, but have only a traditional exposure to elementary algebra. Secondary school and college teachers and students who want to expand their horizons in the field will find a fresh presentation of familiar concepts and some unexpected results. This book serves as a text for an appreciation course in modern mathematics designed for non-mathematics majors or for first-year students who are considering the possibility of studying mathematics or related disciplines. It can also serve as a source of computer-supported activities that could supplement traditional courses in algebra, multivariable calculus, and complex variable. This book gives the reader a sense of the visual nature of mathematics. Mathematical experiments with universal mapping software VisuMatica, designed by Vladimir Nodel'man, form the very core of the book. Readers are encouraged to reproduce, play with, and expand on these experiments. Numerous problems are interspersed throughout the text to guide the reader. Our treatment of standard algebra is visual and computational. By introducing visual computational environments like VisuMatica, our book promotes this geometric approach to algebra and makes it accessible to readers a great deal earlier. The book will enable our readers to approach its content on three levels: the first one which requires only some fluency with elementary algebraic manipulations; the second one which also presumes familiarity with the notions of derivatives and tangent lines to plane curves, and the third one which uses some basic concepts of multivariable calculus. All three levels are clearly marked in the text, and will allow for a smooth reading and virtual experiments, regardless of the level that our readers will find comfortable.

shape algebra: *Relational and Algebraic Methods in Computer Science* Harrie de Swart,

2011-06-07 This book constitutes the proceedings of the 12 International Conference on Relational and Algebraic Methods in Computer Science, RAMICS 2011, held in Rotterdam, The Netherlands, in May/June 2011. This conference merges the RelMICS (Relational Methods in Computer Science) and AKA (Applications of Kleene Algebra) conferences, which have been a main forum for researchers who use the calculus of relations and similar algebraic formalisms as methodological and conceptual tools. Relational and algebraic methods and software tools turn out to be useful for solving problems in social choice and game theory. For that reason this conference included a special track on Computational Social Choice and Social Software. The 18 papers included were carefully reviewed and selected from 27 submissions. In addition the volume contains 2 invited tutorials and 5 invited talks.

shape algebra: Design Computing and Cognition'20 John S. Gero, 2022-02-24 The papers in this volume are from the Ninth International Conference on Design Computing and Cognition (DCC'20) held virtually at the Georgia Institute of Technology, Atlanta, USA. They represent the state-of-the-art of research and development in design computing and design cognition including the increasingly active area of design cognitive neuroscience. They are of particular interest to design researchers, developers and users of advanced computation in designing as well as to design educators. This volume contains knowledge about the cognitive behavior of designers, which is valuable for those who need to gain a better understanding of designing.

shape algebra: Algebra - Representation Theory Klaus W. Roggenkamp, Mirela Stefanescu, 2001-08-31 Over the last three decades representation theory of groups, Lie algebras and associative algebras has undergone a rapid development through the powerful tool of almost split sequences and the Auslander-Reiten quiver. Further insight into the homology of finite groups has illuminated their representation theory. The study of Hopf algebras and non-commutative geometry is another new branch of representation theory which pushes the classical theory further. All this can only be seen in connection with an understanding of the structure of special classes of rings. The aim of this book is to introduce the reader to some modern developments in: Lie algebras, quantum groups, Hopf algebras and algebraic groups; non-commutative algebraic geometry; representation theory of finite groups and cohomology; the structure of special classes of rings.

shape algebra: The Electronic Design Studio Malcolm McCullough, William John Mitchell, Patrick Purcell, 1990 In four parts this book frames those issues and provides a diversity of perspectives on them.

shape algebra: *Understanding and Teaching Primary Mathematics* Tony Cotton, 2020-09-02 Written by an experienced teacher and teacher educator with widespread experience of teaching mathematics in the UK and internationally, Understanding and Teaching Primary Mathematics combines pedagogy and subject knowledge to build confidence and equip you with all the skills and know-how you need to successfully teach mathematics to children of any age. This fourth edition has been fully updated to reflect the latest research developments and initiatives in the field, including a brand-new chapter on 'Mastery and mathematics' and 'The Singapore approach' which reflects the current international interest in these approaches to learning and teaching mathematics. Extra features also include helpful callouts to the book's revised and updated companion website, which offers a shared site with a range of resources relevant to both this book and its companion volume, Teaching for Mathematical Understanding. Stimulating, accessible and well-illustrated, with comprehensive coverage of subject knowledge and pedagogy, Understanding and Teaching Primary Mathematics is an essential purchase for trainee and practising teachers alike.

shape algebra: How Chinese Learn Mathematics Lianghuo Fan, 2004 The book has been written by an international group of very activeresearchers and scholars who have a passion for the study of Chinesemathematics education. It aims to provide readers with a comprehensiveand updated picture of the teaching and learning of mathematicsinvolving Chinese students from various perspectives, including theways in which Chinese students learn mathematics in classrooms, schools and homes, the influence of the cultural and socialenvironment on Chinese students'' mathematics learning, and thestrengths and weaknesses of the ways in which Chinese learnmathematics

shape algebra: Design Computing and Cognition’22 John S Gero, 2023-01-04 This book reports research and development that represent the state of the art in artificial intelligence in design, design cognition, design neurocognition, and design theories from the Tenth International Conference on Design Computing and Cognition held in Glasgow, UK, in 2022. The 48 chapters are grouped under the headings of natural language processing and design; design cognition; design neurocognition; learning and design; creative design and co-design; shape grammars; quantum computing; and human behavior. These contributions are of particular interest to design researchers and design educators, as well as to users of advanced computation and cognitive science. This book contains knowledge about the cognitive and neurocognitive behavior of designers, which is valuable to those who need to gain a better understanding of designing.

shape algebra: Standards-Driven Math Vocabulary Ranking Nathaniel Rock, 2005-08 A textbook and classroom supplement for students, parents, teachers, and administrators who need better options for math intervention classes ranging in difficulty from pre-algebra to geometry. Included are more than 750 middle school and high school math vocabulary words ranked in order from easiest to hardest for maximum standards-driven, informed, intervention instruction. (Mathematics)

shape algebra: Design Computing and Cognition '18 John S. Gero, 2019-01-07 This is the proceedings of the Eighth International Conference on Design Computing and Cognition (DCC’18) held at the Polytecnico di Milano in Italy. This volume presents both advances in theory and applications and demonstrates the depth and breadth of design computing and design cognition. Design thinking, the label given to the acts of designing, has become a paradigmatic view that has transcended the discipline of design and is now widely used in business and elsewhere. As a consequence there is an increasing interest in design research. This volume contains papers that represent the state-of-the-art research and developments in design computing and design cognition. This book is of particular interest to researchers, developers and users of advanced computation in design and those who need to gain a better understanding of designing that can be obtained through empirical studies.

shape algebra: Geometric Transitions Jeffrey Danciger, 2011 We introduce a geometric transition between two homogeneous three-dimensional geometries: hyperbolic geometry and anti de Sitter (AdS) geometry. Given a path of three-dimensional hyperbolic structures that collapse down onto a hyperbolic plane, we describe a method for constructing a natural continuation of this path into AdS structures. In particular, when hyperbolic cone manifolds collapse, the AdS manifolds generated on the other side of the transition have tachyon singularities. The method involves the study of a new transitional geometry called half-pipe geometry. We also discuss combinatorial/algebraic tools for constructing transitions using ideal tetrahedra. Using these tools we prove that transitions can always be constructed when the underlying manifold is a punctured torus bundle.

shape algebra: Shapes of Imagination George Stiny, 2022-11-15 Visual calculating in shape grammars aligns with art and design, bridging the gap between seeing (Coleridge's “imagination”) and combinatoric play (Coleridge's “fancy”). In Shapes of Imagination, George Stiny runs visual calculating in shape grammars through art and design—incorporating Samuel Taylor Coleridge's poetic imagination and Oscar Wilde's corollary to see things as they aren't. Many assume that calculating limits art and design to suit computers, but shape grammars rely on seeing to prove otherwise. Rules that change what they see extend calculating to overtake what computers can do, in logic and with data and learning. Shape grammars bridge the divide between seeing (Coleridge's “imagination, or esemplastic power”) and combinatoric play (Coleridge's “fancy”). Stiny shows that calculating without seeing excludes art and design. Seeing is key for calculating to augment creative activity with aesthetic insight and value. Shape grammars go by appearances, in a full-fledged aesthetic enterprise for the inconstant eye; they answer the question of what calculating would be like if Turing and von Neumann were artists instead of logicians. Art and design are calculating in all their splendid detail.

shape algebra: Threading Homology Through Algebra G. Boffi, David Buchsbaum, 2006-06-29 Aimed at graduate students and researchers in mathematics, this book takes homological themes, such as Koszul complexes and their generalizations, and shows how these can be used to clarify certain problems in selected parts of algebra, as well as their success in solving a number of them.

shape algebra: *Design Computing and Cognition '04* Asko Riitahuhta, 2013-06-29 Artificial intelligence provides an environmentally rich paradigm within which design research based on computational constructions can be carried out. This has been one of the foundations for the developing field called design computing. Recently, there has been a growing interest in what designers do when they design and how they use computational tools. This forms the basis of a newly emergent field called design cognition that draws partly on cognitive science. This new conference series aims to provide a bridge between the two fields of design computing and design cognition. The papers in this volume are from the First International Conference on Design Computing and Cognition (DCC'04) held at the Massachusetts Institute of Technology, USA. They represent state-of-the art research and development in design computing and cognition. They are of particular interest to researchers, developers and users of advanced computation in design and those who need to gain a better understanding of designing.

shape algebra: History of Topology I.M. James, 1999-08-24 Topology, for many years, has been one of the most exciting and influential fields of research in modern mathematics. Although its origins may be traced back several hundred years, it was Poincaré who gave topology wings in a classic series of articles published around the turn of the century. While the earlier history, sometimes called the prehistory, is also considered, this volume is mainly concerned with the more recent history of topology, from Poincaré onwards.As will be seen from the list of contents the articles cover a wide range of topics. Some are more technical than others, but the reader without a great deal of technical knowledge should still find most of the articles accessible. Some are written by professional historians of mathematics, others by historically-minded mathematicians, who tend to have a different viewpoint.

shape algebra: Design Computing and Cognition '12 John S. Gero, 2014-09-03 Design thinking, the label given to the acts of designing, has become a paradigmatic view that has transcended the discipline of design and is now widely used in business and elsewhere. As a consequence there is an increasing interest in design research. This is because of the realization that design is part of the wealth creation of a nation and needs to be better understood and taught. The continuing globalization of industry and trade has required nations to re-examine where their core contributions lie if not in production efficiency. Design is a precursor to manufacturing for physical objects and is the precursor to implementation for virtual objects. At the same time, the need for sustainable development requires the design of new products and processes, which feeds a movement towards design innovations and inventions. The papers in this volume are from the Fifth International Conference on Design Computing and Cognition (DCC'12) held at Texas A & M University, USA. They represent the state-of-the-art of research and development in design computing and design cognition. They are of particular interest to researchers, developers and users of advanced computation in design and those who need to gain a better understanding of designing.

Related to shape algebra

python - [0] vs x [0].shape in NumPy - Stack Overflow 10 x[0].shape will give the Length of 1st row of an array. x.shape[0] will give the number of rows in an array. In your case it will give output 10. If you will type x.shape[1], it will

arrays - what does numpy ndarray shape do? - Stack Overflow 82 yourarray.shape or np.shape() or np.ma.shape() returns the shape of your ndarray as a tuple; And you can get the (number of) dimensions of your array using

tensorflow placeholder - understanding `shape= [None,` You can think of a placeholder in TensorFlow as an operation specifying the shape and type of data that will be fed into the

graph.placeholder X defines that an unspecified number of rows of

python - ValueError: shape mismatch: objects cannot be broadcast ValueError: shape mismatch: objects cannot be broadcast to a single shape It computes the first two (I am running several thousand of these tests in a loop) and then dies

android - How to set shape's opacity? - Stack Overflow I already know how to set the opacity of the background image but I need to set the opacity of my shape object. In my Android app, I have it like this: and I want to make this black

What does shape[0] and shape[1] do in python? - Stack Overflow In python shape [0] returns the dimension but in this code it is returning total number of set. Please can someone tell me work of shape [0] and shape [1]? Code: m_train =

python - shape vs len for numpy array - Stack Overflow Still, performance-wise, the difference should be negligible except for a giant giant 2D dataframe. So in line with the previous answers, df.shape is good if you need both

python - ValueError: could not broadcast input array from shape ValueError: could not broadcast input array from shape (224,224,3) into shape (224) But the following will work, albeit with different results than (presumably) intended

shape = 19, shape = 20 and shape=16 in R graphics [duplicate] In R graphics and ggplot2 we can specify the shape of the points. I am wondering what is the main difference between shape = 19, shape = 20 and shape = 16? Is it the size?

OpenCV shape matching between two similar shapes OpenCV shape matching between two similar shapes Asked 6 years, 6 months ago Modified 3 years, 1 month ago Viewed 32k times

python - [0] vs x [0].shape in NumPy - Stack Overflow 10 x[0].shape will give the Length of 1st row of an array. x.shape[0] will give the number of rows in an array. In your case it will give output 10. If you will type x.shape[1], it will

arrays - what does numpy ndarray shape do? - Stack Overflow 82 yourarray.shape or np.shape() or np.ma.shape() returns the shape of your ndarray as a tuple; And you can get the (number of) dimensions of your array using

tensorflow placeholder - understanding `shape= [None,` You can think of a placeholder in TensorFlow as an operation specifying the shape and type of data that will be fed into the graph.placeholder X defines that an unspecified number of rows of

python - ValueError: shape mismatch: objects cannot be broadcast ValueError: shape mismatch: objects cannot be broadcast to a single shape It computes the first two (I am running several thousand of these tests in a loop) and then dies

android - How to set shape's opacity? - Stack Overflow I already know how to set the opacity of the background image but I need to set the opacity of my shape object. In my Android app, I have it like this: and I want to make this black

What does shape[0] and shape[1] do in python? - Stack Overflow In python shape [0] returns the dimension but in this code it is returning total number of set. Please can someone tell me work of shape [0] and shape [1]? Code: m_train =

python - shape vs len for numpy array - Stack Overflow Still, performance-wise, the difference should be negligible except for a giant giant 2D dataframe. So in line with the previous answers, df.shape is good if you need both

python - ValueError: could not broadcast input array from shape ValueError: could not broadcast input array from shape (224,224,3) into shape (224) But the following will work, albeit with different results than (presumably) intended

shape = 19, shape = 20 and shape=16 in R graphics [duplicate] In R graphics and ggplot2 we can specify the shape of the points. I am wondering what is the main difference between shape = 19, shape = 20 and shape = 16? Is it the size?

OpenCV shape matching between two similar shapes OpenCV shape matching between two similar shapes Asked 6 years, 6 months ago Modified 3 years, 1 month ago Viewed 32k times

python - [0] vs x [0].shape in NumPy - Stack Overflow 10 x[0].shape will give the Length of 1st

row of an array. x.shape[0] will give the number of rows in an array. In your case it will give output 10. If you will type x.shape[1], it will

arrays - what does numpy ndarray shape do? - Stack Overflow 82 yourarray.shape or np.shape() or np.ma.shape() returns the shape of your ndarray as a tuple; And you can get the (number of) dimensions of your array using

tensorflow placeholder - understanding `shape= [None,` You can think of a placeholder in TensorFlow as an operation specifying the shape and type of data that will be fed into the graph.placeholder X defines that an unspecified number of rows of

python - ValueError: shape mismatch: objects cannot be broadcast ValueError: shape mismatch: objects cannot be broadcast to a single shape It computes the first two (I am running several thousand of these tests in a loop) and then dies

android - How to set shape's opacity? - Stack Overflow I already know how to set the opacity of the background image but I need to set the opacity of my shape object. In my Android app, I have it like this: and I want to make this black

What does shape[0] and shape[1] do in python? - Stack Overflow In python shape [0] returns the dimension but in this code it is returning total number of set. Please can someone tell me work of shape [0] and shape [1]? Code: m_train =

python - shape vs len for numpy array - Stack Overflow Still, performance-wise, the difference should be negligible except for a giant giant 2D dataframe. So in line with the previous answers, df.shape is good if you need both

python - ValueError: could not broadcast input array from shape ValueError: could not broadcast input array from shape (224,224,3) into shape (224) But the following will work, albeit with different results than (presumably) intended

shape = 19, shape = 20 and shape=16 in R graphics [duplicate] In R graphics and ggplot2 we can specify the shape of the points. I am wondering what is the main difference between shape = 19, shape = 20 and shape = 16? Is it the size?

OpenCV shape matching between two similar shapes OpenCV shape matching between two similar shapes Asked 6 years, 6 months ago Modified 3 years, 1 month ago Viewed 32k times

python - [0] vs x [0].shape in NumPy - Stack Overflow 10 x[0].shape will give the Length of 1st row of an array. x.shape[0] will give the number of rows in an array. In your case it will give output 10. If you will type x.shape[1], it will

arrays - what does numpy ndarray shape do? - Stack Overflow 82 yourarray.shape or np.shape() or np.ma.shape() returns the shape of your ndarray as a tuple; And you can get the (number of) dimensions of your array using

tensorflow placeholder - understanding `shape= [None,` You can think of a placeholder in TensorFlow as an operation specifying the shape and type of data that will be fed into the graph.placeholder X defines that an unspecified number of rows of

python - ValueError: shape mismatch: objects cannot be broadcast ValueError: shape mismatch: objects cannot be broadcast to a single shape It computes the first two (I am running several thousand of these tests in a loop) and then dies

android - How to set shape's opacity? - Stack Overflow I already know how to set the opacity of the background image but I need to set the opacity of my shape object. In my Android app, I have it like this: and I want to make this black

What does shape[0] and shape[1] do in python? - Stack Overflow In python shape [0] returns the dimension but in this code it is returning total number of set. Please can someone tell me work of shape [0] and shape [1]? Code: m_train =

python - shape vs len for numpy array - Stack Overflow Still, performance-wise, the difference should be negligible except for a giant giant 2D dataframe. So in line with the previous answers, df.shape is good if you need both

python - ValueError: could not broadcast input array from shape ValueError: could not broadcast input array from shape (224,224,3) into shape (224) But the following will work, albeit

with different results than (presumably) intended

shape = 19, shape = 20 and shape=16 in R graphics [duplicate] In R graphics and ggplot2 we can specify the shape of the points. I am wondering what is the main difference between shape = 19, shape = 20 and shape = 16? Is it the size?

OpenCV shape matching between two similar shapes OpenCV shape matching between two similar shapes Asked 6 years, 6 months ago Modified 3 years, 1 month ago Viewed 32k times

python - [0] vs x [0].shape in NumPy - Stack Overflow 10 x[0].shape will give the Length of 1st row of an array. x.shape[0] will give the number of rows in an array. In your case it will give output 10. If you will type x.shape[1], it will

arrays - what does numpy ndarray shape do? - Stack Overflow 82 yourarray.shape or np.shape() or np.ma.shape() returns the shape of your ndarray as a tuple; And you can get the (number of) dimensions of your array using

tensorflow placeholder - understanding `shape= [None,` You can think of a placeholder in TensorFlow as an operation specifying the shape and type of data that will be fed into the graph.placeholder X defines that an unspecified number of rows of

python - ValueError: shape mismatch: objects cannot be broadcast ValueError: shape mismatch: objects cannot be broadcast to a single shape It computes the first two (I am running several thousand of these tests in a loop) and then dies

android - How to set shape's opacity? - Stack Overflow I already know how to set the opacity of the background image but I need to set the opacity of my shape object. In my Android app, I have it like this: and I want to make this black

What does shape[0] and shape[1] do in python? - Stack Overflow In python shape [0] returns the dimension but in this code it is returning total number of set. Please can someone tell me work of shape [0] and shape [1]? Code: m_train =

python - shape vs len for numpy array - Stack Overflow Still, performance-wise, the difference should be negligible except for a giant giant 2D dataframe. So in line with the previous answers, df.shape is good if you need both

python - ValueError: could not broadcast input array from shape ValueError: could not broadcast input array from shape (224,224,3) into shape (224) But the following will work, albeit with different results than (presumably) intended

shape = 19, shape = 20 and shape=16 in R graphics [duplicate] In R graphics and ggplot2 we can specify the shape of the points. I am wondering what is the main difference between shape = 19, shape = 20 and shape = 16? Is it the size?

OpenCV shape matching between two similar shapes OpenCV shape matching between two similar shapes Asked 6 years, 6 months ago Modified 3 years, 1 month ago Viewed 32k times

python - [0] vs x [0].shape in NumPy - Stack Overflow 10 x[0].shape will give the Length of 1st row of an array. x.shape[0] will give the number of rows in an array. In your case it will give output 10. If you will type x.shape[1], it will

arrays - what does numpy ndarray shape do? - Stack Overflow 82 yourarray.shape or np.shape() or np.ma.shape() returns the shape of your ndarray as a tuple; And you can get the (number of) dimensions of your array using

tensorflow placeholder - understanding `shape= [None,` You can think of a placeholder in TensorFlow as an operation specifying the shape and type of data that will be fed into the graph.placeholder X defines that an unspecified number of rows of

python - ValueError: shape mismatch: objects cannot be broadcast ValueError: shape mismatch: objects cannot be broadcast to a single shape It computes the first two (I am running several thousand of these tests in a loop) and then dies

android - How to set shape's opacity? - Stack Overflow I already know how to set the opacity of the background image but I need to set the opacity of my shape object. In my Android app, I have it like this: and I want to make this black

What does shape[0] and shape[1] do in python? - Stack Overflow In python shape [0] returns

the dimension but in this code it is returning total number of set. Please can someone tell me work of shape [0] and shape [1]? Code: m_train =

python - shape vs len for numpy array - Stack Overflow Still, performance-wise, the difference should be negligible except for a giant giant 2D dataframe. So in line with the previous answers, df.shape is good if you need both

python - ValueError: could not broadcast input array from shape ValueError: could not broadcast input array from shape (224,224,3) into shape (224) But the following will work, albeit with different results than (presumably) intended

shape = 19, shape = 20 and shape=16 in R graphics [duplicate] In R graphics and ggplot2 we can specify the shape of the points. I am wondering what is the main difference between shape = 19, shape = 20 and shape = 16? Is it the size?

OpenCV shape matching between two similar shapes OpenCV shape matching between two similar shapes Asked 6 years, 6 months ago Modified 3 years, 1 month ago Viewed 32k times

Related to shape algebra

Newly discovered 'einstein' tile is a 13-sided shape that solves a decades-old math problem (Live Science2y) A new 13-sided shape is the first example of an elusive "einstein" — a single shape that can be tiled infinitely without repeating a pattern. When you purchase through links on our site, we may earn

Newly discovered 'einstein' tile is a 13-sided shape that solves a decades-old math problem (Live Science2y) A new 13-sided shape is the first example of an elusive "einstein" — a single shape that can be tiled infinitely without repeating a pattern. When you purchase through links on our site, we may earn

Texas’ Algebra II debate could shape school policy (Washington Examiner12y) The math class of exponents and imaginary numbers — and a potential predictor of a student’s success in college and in life — has become a key point of contention as the Texas Legislature grapples

Texas’ Algebra II debate could shape school policy (Washington Examiner12y) The math class of exponents and imaginary numbers — and a potential predictor of a student’s success in college and in life — has become a key point of contention as the Texas Legislature grapples

Back to Home: http://www.speargroupllc.com